DOCUMENT RESUME

ED 281 716 SE 047 910

AUTHOR Marshall, Sandra P.; And Others
TITLE Schema Knowledge Structures for Representing and Understanding Arithmetic Story Problems. First Year Technical Report.
INSTITUTION San Diego State Univ., Calif. Dept. of Psychology.
SPONS AGENCY Office of Naval Research, Arlington, Va.
PUB DATE Mar 87
CONTRACT N-00014-K-85-0661
NOTE 72p.
PUB TYPE Reports - Research/Technical (143)

EDRS PRICE MF01/PC03 Plus Postage.
DESCRIPTORS *Cognitive Processes; *Computer Simulation; Educational Research; Elementary Education; *Elementary School Mathematics; Learning Theories; *Long Term Memory; *Mathematics Instruction; Memory; Models; *Problem Solving; Research Reports; Schemata (Cognition)
IDENTIFIERS *Mathematics Education Research

ABSTRACT

This report describes the first of three stages in a study in the domain of problem solving: the definition and explication of schema knowledge. One objective of this research is to understand how schema knowledge is acquired and used in the chosen domain. The focus is on ways in which instruction influences the development of specific knowledge structures in long-term memory. The problem-solving domain was first analyzed with respect to the conceptual relations that may appear in arithmetic story problems. Emphasis is placed on the underlying semantic structure that gives meaning to each problem. Next, each semantic structure was framed as a hypothetical memory object, or schema, and the relationship to accepted theories of memory was explored. Finally, a computer model was developed that details the linkage of schema knowledge to two basic components of long-term memory: semantic networks of declarative memory and production systems and procedural memory (MNS)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

ED281716

SDSU

SCHEMA KNOWLEDGE STRUCTURES FOR REPRESENTING AND UNDERSTANDING ARITHMETIC STORY PROBLEMS

by

Sandra P. Marshall
Christopher A. Pribe
Julie D. Smith

March 1987

Center for Research in Mathematics and Science Education

DEPARTMENT OF PSYCHOLOGY
College of Sciences
San Diego State University
San Diego, CA 92182-0413

This research is supported by the Office of Naval Research, Cognitive Sciences Programs, Contract No. N-00014-K-85-0661.

Reproduction in whole or part is permitted for any purposes of the United States Government.

Approved for public release; distribution unlimited.

SE 047 910

BEST COPY AVAILABLE

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION: unclassified	1b. RESTRICTIVE MARKINGS: none
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT: ONR distribution list; availability unlimited
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION: San Diego State Univ. / 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: Cognitive Sciences Programs, Office of Naval Research
6c. ADDRESS (City, State, and ZIP Code): Psychology Department, San Diego, CA 92182	7b. ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING / SPONSORING ORGANIZATION: Office of Naval Research / 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: N-00014-K-85-0661
8c. ADDRESS (City, State, and ZIP Code): Cognitive Sciences Programs (Code 1142 PT), Arlington, VA 22217	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO; PROJECT NO N00014 K-85-0661; TASK NO; WORK UNIT ACCESSION NO

11. TITLE (Include Security Classification): Schema Knowledge Structures for Representing and Understanding Arithmetic Story Problems

12. PERSONAL AUTHOR(S): Sandra P. Marshall, Christopher A. Pribe, & Julie D. Smith

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Technical	FROM 9-1-85 TO 9-1-86	1987, March	67

16. SUPPLEMENTARY NOTATION: Approved for public release; distribution unlimited

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	problem solving, cognitive processes, mathematics instruction, knowledge representation

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The research described in this report is part of a three-stage project in the domain of arithmetic story problems. The three stages are: (1) definition and explication of schema knowledge, (2) development and evaluation of an instructional system designed to teach schema knowledge, and (3) computer simulation of the acquisition and use of schema knowledge structures. This document focuses on the first stage only and explicates three aspects of our research. First, the domain itself is analyzed with respect to the conceptual relations that may appear in story problems. Emphasis is upon the underlying semantic structure that gives meaning to each problem. Second, each semantic structure is framed as a hypothetical memory object (i.e., a schema), and the relationship to accepted theories of memory is explored. Third, a computer model is

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT: ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL: Susan Chipman	22b. TELEPHONE (Include Area Code): (202) 696-4318 / 22c. OFFICE SYMBOL

DD Form 1473, JUN 86 — Previous editions are obsolete. — SECURITY CLASSIFICATION OF THIS PAGE: Unclassified

presented that details the linkage of schema knowledge to two basic components of long-term memory: semantic networks of declarative memory and production systems of procedural memory.

Schema Knowledge Structures for Representing and Understanding Arithmetic Story Problems

First Year Technical Report
ONR Contract N00014-85-K-0661

Sandra P. Marshall
Christopher A. Pribe
Julie D. Smith

Department of Psychology
San Diego State University
San Diego, CA 92182

Acknowledgments

We would like to thank Kathryn Barthuli and Michael Kummle for their assistance in preparing this report.

Introduction

The research described in this report is part of a three-stage project in the domain of arithmetic story problems. The three stages are:

(1) definition and explication of schema knowledge

(2) development and evaluation of an instructional system designed to teach schema knowledge

(3) computer simulation of the acquisition and use of schema knowledge structures.

This document focuses on the first stage only. The remaining stages will be addressed in future reports.

One objective of this research is to understand how schema knowledge is acquired and used in the chosen domain. In particular, the focus is upon ways in which instruction influences the development of specific knowledge structures in long-term memory.

It is common to find schema-based research in cognitive science. Much of this work, however, fails to specify precisely the nature of a schema. A schema has sometimes been considered to be equivalent to other cognitive structures such as frame, script, or plan. Most often, a schema is no more than a simple declarative frame with variable slots to be filled. The lack of specificity about the structure of a schema makes it difficult to describe or model how an individual learns, stores, and uses knowledge. If the schema is the basic building block of cognition, as Rumelhart (1980) states, then it must be more clearly defined. This report focuses on the definition and structure of schema knowledge. The adequacy of the definition is evaluated through computer simulation programs that operate within the domain of interest.

The following aspects of our research are described below. First, the domain itself is subjected to analysis of the conceptual relations that may be expressed in arithmetic story problems. Emphasis is placed upon the underlying semantic structure that gives meaning to each problem. Second, each semantic structure is framed as a hypothetical memory object (i.e., a schema), and the relationship to accepted theories of memory is developed. Third, a computer model is presented that details the linkage of schema knowledge to two basic components of long-term memory: semantic networks of declarative memory and production systems of procedural memory.

Semantic Relations of Arithmetic Story Problems

A story problem may be loosely characterized as an abbreviated verbal account of a situation, providing some specific information (usually numerical) and requiring use of that information to answer a stated question. The information given in the problem is embedded within a "story", but the story rarely contains much detail. The reader is expected to recognize the situation depicted in the story and to embellish it from his or her own store of experiences. Thus, it becomes important that the reader have sufficient knowledge stored in memory to be used as necessary in understanding the events of the story.

What is required to understand a story problem? First, the reader must recognize the words used in the problem. Second, the reader must understand the situation and the relationships that exist among objects described in the problem. We do not focus here on the problems of reading; it is assumed that students have the requisite knowledge of the situations portrayed in the problems. Our attention is upon what knowledge is required for students to understand the abbreviated description of the situation and its accompanying relations.

One aspect of understanding appears to be the categorizing of similar items. Recognition then becomes the process identifying the appropriate category. It is obvious that one could organize the domain of story problems on a number of different dimensions. For example, problems requiring the same operation(s) could be grouped together. Alternatively, problems describing the same situation(s) could be aligned. Or, as we suggest here, problems reflecting similar semantic structure could be aggregated.

Categorization by operation works well only when a single operation is required. This approach is currently popular with teachers and textbook developers, but it has limited success as a problem-solving strategy. For multi-step problems, it is difficult for students to order and keep track of the various operations.

Grouping by story situation also has its drawbacks. There are an infinite number of situations that could be used, and memory requirements for keeping track of all of them would be enormous. Also, for any situation, several problems could be devised, each requiring different methods of solution. Hence, the situation alone could not sufficiently provide clues about solution strategy.

We argue here that the most efficient and successful means of organization is to look for common underlying

elements within the structure of the problems. This approach involves features that are usually termed semantic relations. It requires the ability to understand the situation depicted in the story and to perceive the relationships that exist between objects.

In this section, we describe a set of semantic relations that characterize fully the domain of arithmetic story problems. Other studies in this domain have either limited their scope to a subset of arithmetic operations (e.g., Riley, Greeno, & Heller, 1983; Briars & Larkin, 1984) or have focused upon characteristics of the quantities given in the problem rather than upon the overall structure of the problem (cf. Greeno, Brown, Foss, Shalin, Bee, Lewis, & Vitolo, 1986). Differences between the current conceptualization and those of other researchers, particularly Riley et al. and Greeno et al., will be discussed later in some detail.

Five semantic relations appear to be sufficient for characterizing virtually all story problems of arithmetic. These are: Change, Combine, Compare, Vary, and Transform. Each is described below, together with examples that demonstrate the different formulations of problems containing the relations.

The CHANGE Relation

The first and most elementary of the semantic relations is the Change (CH) relation. In its simplest form, the relation is an increment or decrement of a measurable resource. The change that occurs is physical and is permanent. Once the change occurs, the original state cannot be revisited.

A CH relation can be manifested in many forms. The following are examples:

> John had 12 baseball cards. His friend Jim gave him 5 more. How many cards does John have now? (1)
>
> Twenty-five tomato plants were growing in the garden. Snails ate some of them. There are 15 plants left. How many did the snails eat? (2)
>
> I had some money in my checking account. After I deposited a check for $35.00, I had $280.75 in the account. How much was in the account before I made the deposit? (3)

Each problem begins with an initial state (e.g., 12 baseball cards, 25 tomato plants, some money in an account) in which

the objects to be manipulated are specified and in which the quantity or amount of these objects is defined. A change in possession occurs at some later time, indicated in these problems by the phrases or words "more" (than he had before), "left" (after eating), and "after" (the check was deposited). The reader must infer the time constraints from the position of the statements and these phrases. After the change occurs, there is a recognizable final or ending state in which the new quantity is defined. The initial state and final state are not coexistent: they cannot occur at the same time. Either there are 25 tomato plants or there are 15. Both statements cannot be true.

As shown in examples (1), (2), and (3), three different questions can be posed in a Change relation. The most common situation is to provide information about the initial state and the amount of change, leaving the final state to be computed (1). A second alternative is to present both the initial and final state and to have the individual calculate the amount of change (2). Finally, the problem may contain the amount of change and the final state, and the question is to determine a value for the initial state (3).

Most CH problems involve only additive change (e.g., require addition and subtraction operations). However, sophisticated problems exist for which there may be multiplicative or exponential growth rather than additive change. The following is an example:

> There were 300 bacteria on the petri dish. If they doubled in number every 2 hours, how many would be on the dish after 6 hours? (4)

This example emphasizes an important point about the semantic relations defined here: They are not defined in terms of the arithmetic operations required for problem solution. We are looking at the nature of the relation among objects described within each problem; that relation may itself be linked to several possible operations. Identifying the relation is not synonymous with identifying the operation for solution.

The COMBINE Relation

A COMBINE (CB) relation is expressed whenever there exists a hierarchical or composite grouping of objects within a problem. The CB relation involves the renaming of elements with respect to a superordinate category. No action is taken in a CB relation, and, in contrast to the CH relation, no permanent alteration of objects occurs. The passage of time is irrelevant.

An important characteristic of the CB relation is its dependence upon understanding part-whole relationships. The part-whole concept occurs in many domains,including physics, algebra, and arithmetic (cf. Nesher, Greeno, & Riley, 1982; Kintsch & Greeno, 1985). To understand a Combine relation, an individual must comprehend that the whole (or superordinate category) is equal to the sum of the parts (subordinate categories). A necessary constraint is that the subordinate categories have semantic ties to the superordinate one. The logical, hierarchical structure must have a semantic base. For example, the following item illustrates a Combine relation:

> Ann has three apples and two oranges. How many pieces of fruit does she have? (5)

Consider what an individual must already know (or must be told elsewhere in the problem) in order to solve (5). First, he/she must recognize "apples", "oranges" and "fruit". There must be an understanding of the common attributes of these three elements. If they have no shared characteristics, the problem becomes senseless. The individual must also understand that only those elements specified in the problem are relevant. We do not speculate about how many lemons or grapes are in Ann's possession.

There must also be an awareness of the relationships among apples, oranges, and fruit, as displayed in Figure 1. In this Figure, the two elements "apples" and "oranges" have an identical relationship with the element "fruit": both are instances or examples of "fruit". However, "apples" and "oranges" nonetheless are distinct semantic elements; an apple is not equivalent to an orange. Each element maintains a definitive set of characteristics that serves to distinguish it from other elements existing at the same level (such as pear, grape, lemon). Our hypothetical individual solving the above problem must know that it is logically or semantically impossible to join elements at one level of this semantic network unless they have identical links to a higher level of categorization. That is, they can be combined only if one considers that they are instances of a more general level of classification and only if they equally share the characteristics of that level.

In terms of Figure 1, the elements "apples" and "oranges" inherit the characteristics of "fruit" because there exist links connecting these elements and because "fruit" is at a higher level of the network than the other two elements. The meaning of a link depends upon the direction in which it is interpreted. Thus, it is true that "apples" are an instance of "fruit"; it is not true that "fruit" is an instance of "apples". Similarly, "apples" have

the properties of "fruit"; the converse is not true. In general, links running from superordinate categories to subordinate ones are inheritance links, and it is common to speak of the subordinate or lower level elements inheriting all characteristics of the superordinate one.

The importance of these semantic distinctions is apparent in the following item:

> I have three apples and two oranges. How many pieces of candy do I have? (6)

From Figure 1, it is clear that "fruit" and "candy" are both instances of the superordinate category "food", and both inherit the same set of characteristics about food (e.g., can be eaten, provides source of energy). However, they do not share subordinate elements. That is, "apples" and "oranges" are not examples of "candy" and have no links to it. Thus, problem (6) cannot be solved with the given information.

In general, Combine problems do not define the superordinate and subordinate categories as such. Individuals solving the problems are expected to draw upon semantic knowledge stored in long-term memory for identification and clarification of the elements specified in the problem. If the requisite knowledge is missing from the individual's knowledge base, the individual will be unable to solve the problem.

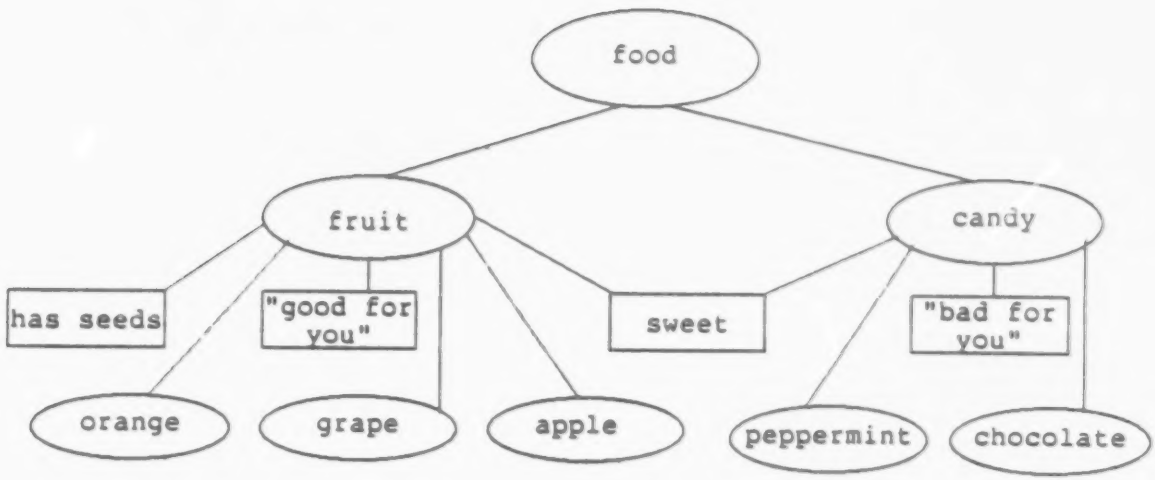
As with the CH relation, there are varying forms that the CB relation may take in an arithmetic story problem. Two different questions may be asked. First, the problem may require finding a numerical value associated with the superordinate category. In that case, values for the relevant subordinate ones must be given in the problem. Second, the problem may ask for the value of one of the subordinate elements. For this case, the superordinate value and the remaining subordinate one must be known. An example of the first situation is given by (5). An example of the second is given below.

> I have three apples and some oranges. If I have five pieces of fruit, how many oranges do I have? (7)

A distinguishing point about the sematic CB relation is that the original quantities associated with the subordinate and superordinate categories remain unchanged by the combination. So, for example, in (7) above, although there may be five pieces of fruit, three of them are still apples.

Figure 1

A Semantic Network



The COMPARE Relation

A third semantic relation in the domain of arithmetic story problems is the Compare (CP) relation. In a CP relation, two elements of a problem are evaluated in order to determine their relative size. The meaning of the relation comes from weighing one element against the other. It is not the absolute value of either one that is central here, but rather the relative position that one has with respect to the other.

A necessary part of the relation is the existence of two elements, each associated with a numerical value and both having the same semantic features. Generally, the semantic features of interest are the units in which the elements are measured (e.g., feet, hours, gallons). Comparisons can only be made meaningfully between elements measured against the same standard. For example, we cannot say which is larger, a liter bottle or a twelve inch board. However, we can compare a liter bottle with an eight ounce jar, provided that we do so on a common unit of measure.

Implicit in the CP relation is the concept of one-to-one matching of one element in the problem with the other. As described by Briars and Larkin (1984), each element is considered to be a set having a given number of members. To compare two sets, one theoretically engages in one-to-one matching, removing one member from each set and setting them apart as a matched pair. The smaller of the two sets is the one which first becomes empty. The amount left in the larger set is the difference between the two sets. If both sets become empty at the same time, they have an equal number of members.

Much like Combine, the Compare relation is static; no action occurs in the problem. The CP relation is a description, an alternative way of expressing the relative size of two sets of similar objects. The representation of time as a variable is usually irrelevant. CP relations can occur at the same time or at different times. For example, consider the following items:

> Joe makes $4.50 per hour at his job, and Ed makes $5.30 per hour. How much more does Joe make per hour than Ed? (8)

> Mary is a gymnast. Last week, she scored 7.5 on the balance beam in a gymnastics meet. In another competition held yesterday, she received a score of 8.5 in the same event. At which meet did she have the best performance? How much better? (9)

In (8), the comparison takes place at a single point in time; both individuals currently make the wages stated in the problem. In (9), the comparison is between scores obtained at two different times. It should be clear from these examples that time is not a distinguishing characteristic of CP.

The VARY Relation

The Compare relation introduced the relationship of relative size. In the Vary (VY) relation, there are several other relationships that must be understood by an individual in order to solve problems having this semantic structure. Most importantly, it is necessary to distinguish between three types of elements: subject-units, object-units, and associations. Subject-units are the foci of the problem (e.g., marbles, apples, children). A subject-unit has a particular object-unit related to it by means of a specified association. For example, the statement that "one apple costs 25 cents" has one apple as the subject-unit, cost as the association, and 25 cents as the object-unit. The association relating subject- and object-units is general and applicable to every subject-unit; thus, any instance of apple will have a cost represented by cents that is associated with it (in the restricted environment of this particular problem, of course).

A second relationship requisite to the VY relation is the concept of per unit. The notion of a constant value per unit may be explicitly stated or may be merely implied by the wording of the problem. In either case, the individual must realize that every instance of the subject-unit presented in the problem will have the same value of an object-unit associated with it. Thus, we have "an apple costs 25 cents" or "the car travels 30 miles on a gallon of gas" as examples. Knowledgeable students understand without being told directly that a second apple will also cost 25 cents and that another gallon of gasoline will enable the car to travel an additional 30 miles. Within a problem, the per-unit value remains constant.

A fundamental difference between VY and the three relations previously defined is that the problem structure of VY involves four quantities; two of these are subject-units and two are object-units. As described above, a subject-unit is paired with an object-unit by means of an association. For the four quantities describing a VY relation, there are two pairs, each bound together by an association. Not only must an individual recognize the pairs and the associations that bind them, the individual must make a mapping from one pair to the other and test the logic of that mapping before solving the problem.

Let the first subject-object pair be denoted by

[subject1-unit <association> object1-unit] (10)

and the second by

[subject2-unit <association> object2-unit]. (11)

Each of the four units of (10) and (11) has two features: a type and a value. The type refers to the nature of the elements, such as apples, pencils, etc. The value is the number of such elements (e.g., 2 apples).

To satisfy the conditions of the VY relation, an individual must test three constraints: that the types of subject-units are identical in expressions (10) and (11), that the types of object-units are identical, and that the associations described in (10) and (11) are the same. Expressions (10) and (11) should differ only in the numerical values associated with the four units. Further, in a typical story problem, three of the four units will have known numerical values. In a VY relation, the objective is to determine the fourth (unknown) value.

This constraint evaluation can best be demonstrated by the following example:

The price of one apple is 25 cents. How much will 15 apples cost? (12)

The expressions of (10) and (11) can be rewritten as:

[1 apple <cost> 25 cents] (13)

and

[15 apples <cost> =?= cents] . (14)

where =?= denotes an unknown value. To solve this problem, an individual first must establish that a logical structure exists. The three tests described above serve this purpose: expressions (13) and (14) both concern apples, both involve the cost of apples, and both measure cost in terms of cents. The importance of these tests is clearly seen by examining the following problems, in which one or more of the tests fail.

The cost of one apple is 25 cents. How much will 5 bananas cost? (15)

The cost of one apple is 25 cents. How much will 15 apples weigh? (16)

The cost of one apple is 25 cents. How much will 5 bananas weigh? (17)

Problems (15) and (16) demonstrate the failure of a single test; problem (17) shows two failures. The more tests that fail, the more illogical the problem appears to an individual. Note that each sentence of (17) is a reasonable statement. The difficulty is that the first cannot be used to answer the question posed in the second.

The TRANSFORM Relation

The fifth and final relation defined here is the Transform (TR) relation. The essential understanding required for a TR is that it is possible to describe one object in several different ways. In particular, if the object has a numerical value associated with it and if it bears a known relationship to another object also having an associated numerical value, one may describe the first object in two ways, in its original metric or as a function of the value of the second object.

Consider a typical Transform problem:

Sue is 1/3 as old as her mother. If her mother is 30 years old, how old is Sue? (18)

There are two ways to look at Sue's age in this problem. First, she must be some number of years old. This is the unknown of the problem. Second, since both her age and her mother's age can be expressed as years, we can look at one as a function of the other. In this case, Sue's age is related to her mother's by the fraction 1/3. Thus, there are two statements about Sue's age and both are simultaneously true.

The TR relation is similar to VY in that two relationships are given in the problem. These may be expressed as:

[subject1 *=* object1-unit] (19)

and

[subject2 *=* object2-unit] (20)

with subject1 and subject2 being the main foci of the problem, with *=* indicating that the leftmost member of the expression can be expressed in terms of the rightmost member, and with object-units having types and values as described above. We also expect to have a known relationship between the two subjects:

[subject1 *=* <f1> subject2]. (21)

This expression states that one subject can be expressed as a function <f1> of the second. Finally, by substitution from (19) and (20) we have

[object1-unit *=* <f1> object2-unit] (22)

that is, the first object-unit can be expressed as a function of the second object-unit (or vice versa).

For a problem to contain a Transform relation, several conditions must be met. First, subject1 and subject2 must be recognizably distinct entities (e.g., two individuals). Second, the two object-units must be expressed in the same metric (e.g., years, feet, days). Third, either the subjects are related to each other through a stated mathematical function or the object-units are so related. Fourth, the problem must provide the value of one object-unit and the value of the mathematical function (such as "3 times as large" or "5 more than"), or it must specify values for both object-units, leaving the mathematical function to be determined.

Using this notation, problem (18) can be represented by the following expressions:

[Sue's age *=* =?= years] (23)

[Mother's age *=* 30 years] (24)

[Sue's age *=* <1/3> Mother's age] (25)

[=?= years *=* <1/3> 30 years] (26)

Expressions (23) and (24) state that both Sue's age and her mother's age can be expressed as some number of years. The number of years is unknown for the former and is given as 30 for the latter. Expression (25) indicates the relationship between the two ages: Sue's age is one-third of her mother's age. Expression (26) is actually the solution to the problem in this case, and it is obtained by substituting the values of the ages from (23) and (24) into the appropriate slots of (25).

An important underlying concept of a TR relation is "unity" or "the whole". Implicit in the definition of relationship between two quantities is the notion that one can frequently define one quantity to be "the whole" and express the second quantity as "a part" or "a multiple" of the whole. This is a more sophisticated use of the part-whole concept than was needed for the Combine relation. Any quantity can be designated "one" or "unity" and any similar quantity can be expressed in relative units. The primary

constraint is that both quantities must be measured by the same metric. In the age problem above, age is given in years. As written, the relation between the two ages is that Sue's age is a fraction of her mother's age. The mother's age is "the whole" and Sue's age is expressed as "a part" of it. Without changing the relationship between ages, one could restate it as "her mother is three times as old as Sue". In this case, Sue's age would be considered the "whole" and the mother's age would be a multiple of it.

The TR relation is static. No action takes place in the problem, there are no alterations in quantities. In fact, the opposite is true: exact conservation of quantities is required in moving from one unit of measure to another. Further, in most TR problems, time is here and now. The relationships that are expressed are true at this moment; they will not necessarily be true at a later date (i.e., consider the age problem above).

A defining characteristic of Transform problems is that one answer to the question posed in the problem is explicitly stated. For (18), the question is "How old is Sue?" One acceptable and true answer is given in the first sentence: "Sue is 1/3 as old as her mother." To force an individual to seek the alternate representation of Sue's age, the problem should state the question as "How many years old is Sue?" An individual solving the problem is expected to know from previous experience that the solution will be expressed in years, even though the explicit question about years is not stated.

There are several ties between TR and other relations. First, both Transform and Combine rely upon the part-whole relationship. Second, both Transform and Compare are concerned with the relative size of quantities. Third, Transform and Vary both involve four quantities, operating on pairs of them.

The TR relation appears to be the most difficult for students to grasp. It is both a prealgebra relation (being fundamental for algebra problem solving) and an arithmetic relation (appearing in story problems as early as third grade arithmetic, CAP 1980). Several constraints must be simultaneously considered in TR. Students may not realize that it is necessary to satisfy many constraints as they seek to recognize the form of a problem; working with a single constraint may lead to an incorrect representation and a consequential incorrect solution. Furthermore, TR is not particularly tied to any arithmetic operation: all four are equally likely.

Multi-Step Problems

Thus far, we have used only simple examples from arithmetic in which a solution may be found by a single application of one arithmetic operation. Most individuals have little trouble with such items, once they have mastered the algorithms of the operations themselves. Rates of success undergo a dramatic shift when problems require more than a single computational step. Even highly qualified students of arithmetic experience difficulty with multi-step problems (Marshall, 1987).

A particular advantage of the semantic relations introduced here is that they may be used to organize multi-step problems. For any problem, there is one central question that is posed and one general situation that is described. Within that situation, there may be other unknowns and other situations that must be examined, but the central one remains the target of the problem solving. For example, we can easily construct a problem in which the central situation is that an individual has some money, makes some purchases, and has a resulting amount of money that is less than the original amount. For example,

> Alice had $50.00 when she went to the grocery store. She bought two quarts of milk at $.75 each, 1 1/2 pounds of cheese at $2.75 per pound, and a loaf of bread for $1.39. How much money did she have after she made these purchases? (27)

This situation expresses a Change relation. Embedded here is a Vary relation (in the case of purchasing more than one item for the same cost per item) and a Combine relation (in the case that the individual purchased several items each having a given price).

Many individuals use an operation-based strategy to solve both simple and difficult problems (Marshall, 1982). Keeping track of the many operations proves to be difficult for a large number of them. We speculate that this difficulty arises because the individuals do not have a means by which they can organize the many steps required in the problem.

It is true that one could read problem (27) and make a mental note that one should **multiply** (2 x $.75), **multiply** (1 1/2 x $2.75), **add** (results of the two multiplications plus $1.39), and **subtract** ($50.00 minus the sum resulting from the addition step). One can hypothesize that an individual attempts to store this list of operations in short-term memory while working the various computations. Since these operations are not logically bound one to the other, some

may become distorted or lost. The result is that one or more of the operations is frequently omitted, yielding an incorrect or partial solution. We have empirical evidence that omission errors are, in fact, among the most common errors on multi-step problems.

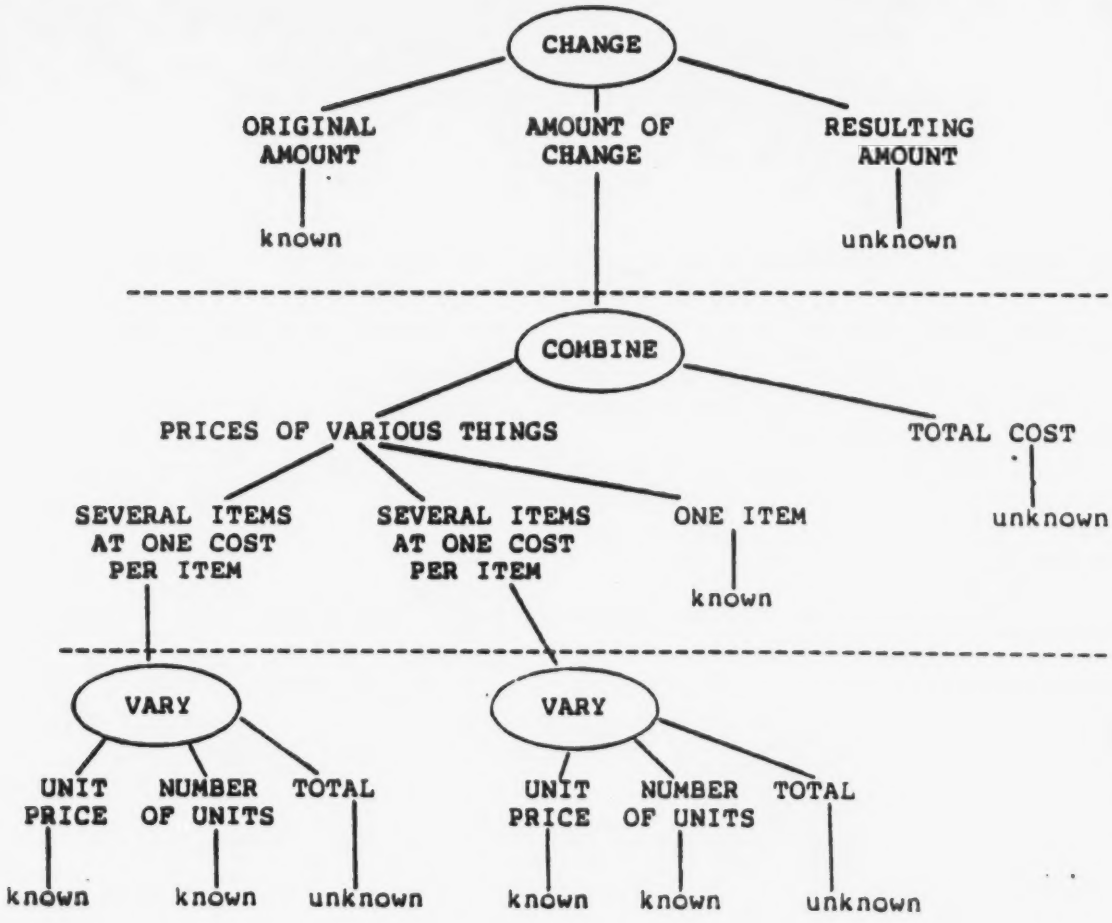
The use of semantic relations introduces a logical structure that serves to organize the information in the problem. In (27), once the individual recognizes that the underlying situation depicts a Change relation, he or she can then look at the components that make up the relation. The initial or starting quantity is already known (i.e., $50.00). The amount of change is not yet known -- this is a secondary problem that must be solved in order to complete the Change relation.

To solve the secondary problem, the individual perceives that several items are to be purchased and that the total cost of these items is required. This represents a Combine relation, and the needed elements are the various prices. Once again, all components of the relation are not known. In this case, there are several items with per-unit prices. Again, this represents a problem within a problem. We are now at the third embedding level. The structure of the problem can be diagrammed as in Figure 2. Using semantic relations in this way imposes a hierarchical structure on the problem-solving steps and thus should enable individuals to monitor these steps.

Figure 2

A Multi-Step Problem

Alice had $50.00 when she went to the grocery store. She bought two quarts of milk at $.75 each, 1 1/2 pounds of cheese at $2.75 per pound, and a loaf of bread for $1.39. How much money did she have after she made these purchases?



The Adequacy of the Classification

The five semantic relations presented above appear to be sufficient for classifying virtually all story problems of arithmetic. Marshall (1985) examined all sixth-grade arithmetic textbooks adopted for use in California public schools. In that study, each story problem was classified according to the relations defined here. All traditional story problems could be uniquely classified. Problems that were not classified were those involving memorized formulas (e.g., find the circumference of a circle, what is the area of the triangle) and were not strictly arithmetic.

For the present study, we evaluated three additional sources of problems, each representing a different level of arithmetic. These were: arithmetic texts for eighth grade, remedial arithmetic materials for community colleges, and a newly created text for training navy personnel. The results were similar to those found by Marshall (1985). Approximately 90% of all items from the three instructional sources could be uniquely classified according to the five semantic relations. The remaining 10% required application of geometric or probability formulas for solution and involved little semantic interpretation. As expected, a somewhat larger number of items involving geometry and probability were found in the present study than were observed at the lower grade because these topics are given greater weight at the upper grades. Therefore, we have a larger proportion of unclassifiable items, labeled "other". The frequencies with which each relation occurred in one, two, and more than two step problems are shown in Table 1.

The two eighth-grade texts contain a total of 629 story problems; 478 (76%) of them illustrate a single relation. The Navy training materials contain only 35 story problems. Most of the problems are simple one-step items (70%). Finally, the remedial materials for community college use had 658 story problems, and 68% of these were single step items as well.

A majority of the items (71%) in these three sources consists of simple single-step story problems that can be solved by application of one arithmetic operation (see Table 1a). A large number of them contain either a Vary or a Transform relation (596 of 935 items, or 64%). Only 15% of all items required use of two different semantic relations (excluding "other"). Also, 18% of the two-step problems were merely repetitions of the same relation. Table 1b contains the frequencies with which various pairings occurred. Finally, a very low 7% of the items contained as many as three semantic relations (see Table 1c).

Table 1

Problem Classification by Textbooks:
The Frequency with Which Semantic Relations Occur

A. One-step problems

Classification	8th Grade Texts	Navy Training Materials	College Remediation	Total
CH	49	0	76	125
CB	31	3	19	53
CP	31	1	47	79
VY	198	9	123	330
TR	135	5	126	266
Other	34	7	41	82
Total	478	25	432	935

B. Two-step problems

Classification	8th Grade Texts	Navy Training Materials	College Remediation	Total
CH/CH	2	0	13	15
CH/CB	1	0	4	5
CH/CP	1	0	1	2
CH/VY	23	1	14	38
CH/TR	23	0	16	39
CH/Other	1	0	1	2
CB/CB	1	1	0	2
CB/CP	3	0	0	3
CB/VY	9	0	33	42
CB/TR	8	0	13	21
CB/Other	1	0	0	1
CP/CP	0	0	1	1
CP/VY	7	1	2	10
CP/TR	0	0	9	9
CP/Other	2	0	2	4
VY/VY	3	0	4	7
VY/TR	6	0	6	12
VY/Other	3	0	10	13
TR/TR	11	0	9	20
TR/Other	4	0	0	4
TOTAL	106	3	138	247

Table 1 continued

C. Problems with more than two steps

	8th Grade Texts	Navy Training Materials	College Remediation	Total
Number of problems	45	7	88	140

D. Number of times a relation occurred in a problem requiring more than two steps **

Classification	8th Grade Texts	Navy Training Materials	College Remediation	Total
CH	29	1	78	108
CB	19	0	16	35
CP	10	7	37	54
VY	32	17	100	149
TR	31	13	42	86
Other	32	0	18	51

** not equivalent to the number of problems

The only problems that we were unable to categorize were items requiring application of formulas. These are represented in Table 1 as "other". In many instances, it would be possible to reduce these also to the semantic information contained in the formula. However, we suspect that most students apply the formulas without deriving them each time as would be necessary if the semantic information were to be used.

We have no reason to believe that these instructional sources are atypical of arithmetic texts in general. The findings are consistent with the classification made at the sixth grade (Marshall, 1985). The evidence is strong that a preponderance of attention is devoted to solving simple story problems at every level of arithmetic.

Other Research About Story Problems

Researchers in the fields of cognitive psychology and mathematics education have developed several approaches to the classification of story problems. These may be loosely grouped into "structure research", in which the underlying relationships are of interest, and "operation research", in which the arithmetic operations themselves are the foci.

Structure Research

Semantic Structure. The major thrust of research about semantic relations has its origins in the work of Riley, Greeno, and Heller (1983). Most recently, this line of research has been extended by Carpenter (1987). Three semantic relations were defined in these studies: Change, Combine, and Compare. These are similar to but not identical with the relations defined here.

Riley et al. opted to study a limited subset of the domain of arithmetic story problems, specifically problems requiring only a single operation of addition or subtraction for solution. This dependence upon operation turns out to be critical. As we pointed out earlier, some of the semantic relations can be present in situations that demand any of the four arithmetic operations (i.e., Transform). By excluding the operations of multiplication and division, one fails to observe the broad structure of the semantic relations. Further, one fails to perceive that semantic relations truly are operation-free.

In the Riley et al. approach, the Change and Combine relations have the same general structure as presented above, but we have defined them with greater specificity and have introduced additional constraints. As Riley et al. pointed out, most of the problems in arithmetic that express

CH and CB relations involve only the operations of addition and subtraction, and their structure is well understood. We have found, however, that there are also multiplicative Change problems, although these are relatively rare. Consequently, we hesitate to use operational labels or constraints.

Our conception of Compare differs substantively from that presented in Riley et al. By their classification, both of the following items demonstrate the Compare relation:

> Joe has 8 marbles. Tom has 5 marbles. How many marbles does Joe have more than Tom? (28)

> Joe has 3 marbles. Tom has 5 marbles more than Joe. How many marbles does Tom have? (29)

Under our theory, only the first of these two items is a Compare problem. The second is an instance of Transform.

We find that the structure of (29) is more similar to that expressed below in (30) than to (28).

> Joe has 3 marbles. Tom has 5 times as many marbles as does Joe. How many marbles does Tom have? (30)

Problems (29) and (30) have the same basic structure. In each case, we know the number of marbles that Joe possesses, and we know that Tom has some number of marbles that can be expressed in terms of Joe's marbles. In both problems, the objective is to use the given relation between the boys' marbles to determine the actual number of marbles owned by Tom. The particular arithmetic operation required is irrelevant to our understanding of the problem.

In this instance, the apparent similarity between (28) and (29) is an artifact of the limited domain. If we work only in the domain of addition and subtraction, Transform problems appear similar to Compare ones because the words "more than" and "less than" appear, just as they do in Compare items. However, these words themselves do not define the Compare relation: it is possible to find them in every semantic relation.

Compositional Structure. A very different conceptualization of structure has been developed by Greeno et al. Under this approach, a problem is characterized by the types of quantities to be found in it (p. 9). Four types of quantities may exist in a problem: extensive, intensive, difference, and factor. Extensive quantities are simply the number of units of some object, such as "5 apples".

Intensive quantities are per unit values, such as "5 words per sentence". A factor is defined as "a unitless quantity that relates two other quantities that have the same units" (p. 9). For example, if "Tom has 3/4 as many marbles as Joe", 3/4 is a factor. Finally, a difference is "an additive relation between two quantities of the same type" (p. 9). In the statement "Tom has 3 more marbles than Joe has", the difference is 3 more.

The importance of these four quantities lies in the ways they may be combined in a problem. Two general compositions are possible: additive compositions and multiplicative compositions. Additive compositions involve only extensive and intensive quantities. Multiplicative compositions may contain all four types of quantities. Greeno et al. specify rules by which compositions may be formed, and they determine the operations used to solve the problems by the types of quantities found in each composition.

A central focus of the research is to make a graphical representation that characterizes the operations. Diagrams are constructed to represent the compositions and the quantities described in the problems. To solve a particular story problem, it is necessary to identify the type of composition and to map the quantities involved in the composition into the appropriate diagram.

Compositional analysis differs significantly from analysis based upon semantic relations. First, compositional analysis has its basis in arithmetic operations (as in additive and multiplicative compositions). Second, the categories of analysis have no meaning when separated from the quantities. That is, general descriptions of situations seem to be irrelevant. In contrast, these general descriptions are used to form the categories of Change, Combine, Compare, Vary, and Transform. Types of numbers or arithmetic operations are secondary to the general description.

Operation Research

Again, there are several different approaches that have been taken. We describe two that seem to be particularly influential and relevant. These are the efforts to classify problems according to their surface features and to classify them according to particular uses or meanings associated with the arithmetic operations.

Surface Structure. There exists a reasonably large body of research devoted to mapping the structural variables that occur in story problems and to assigning difficulty

parameters to these variables. For example, Loftus and Suppes (1972) examined item characteristics such as number of words, the number of sentences, the number of operations, the type of operation, or the similarity to the last-presented item. Multiple regression techniques were used to determine which of these or similar features account for a large proportion of variance in student responses.

While studies such as this one are undeniably interesting in revealing which characteristics of a problem influence the difficulty level of the problem, they nonetheless have little or no direct bearing upon how students learn to solve story problems. The difficulty is that these analyses are based upon external features of the problems having little to do with how individuals understand the problems. Semantic relations are internal features which an individual may relate to knowledge stored in his or her long-term memory.

Use Classes. Usiskin and Bell (1983) give a persuasive argument against employing operations to classify story problems. Their thesis is that operations have more than a single meaning or use. For example, addition may imply both a putting together and a shift. The first of these is closely akin to the semantic relation of Combine. The second corresponds to Change. The value of Usiskin and Bell's approach is that they maintain an emphasis upon operations by changing the focus from algorithms to applications. In so doing, their approach and ours become compatible. They demonstrate the need to evaluate the different uses to which the various operations can be put, and we demonstrate the need to perceive the whole picture as embodied in the semantic relations.

While compatible, the two approaches are not synonymous. Some of the more important differences can be seen in Figure 3. All of the uses defined by Usiskin and Bell for the operations of addition, subtraction, multiplication and division can be mapped into the five semantic relations of Change, Combine, Compare, Vary, and Transform. Most of the uses are evidenced by a single relation. In some cases, a single use might be exemplified in more than a single relation (e.g., the ratio use of division maps into both Compare and Transform).

In the use classification, each operation is expanded into several uses. In the semantic relations classifications, several uses are combined into a smaller number of relations. The important distinction between the use classification and the semantic relations classification is that the latter cuts across arithmetic operations. For example, the Change relation can require addition,

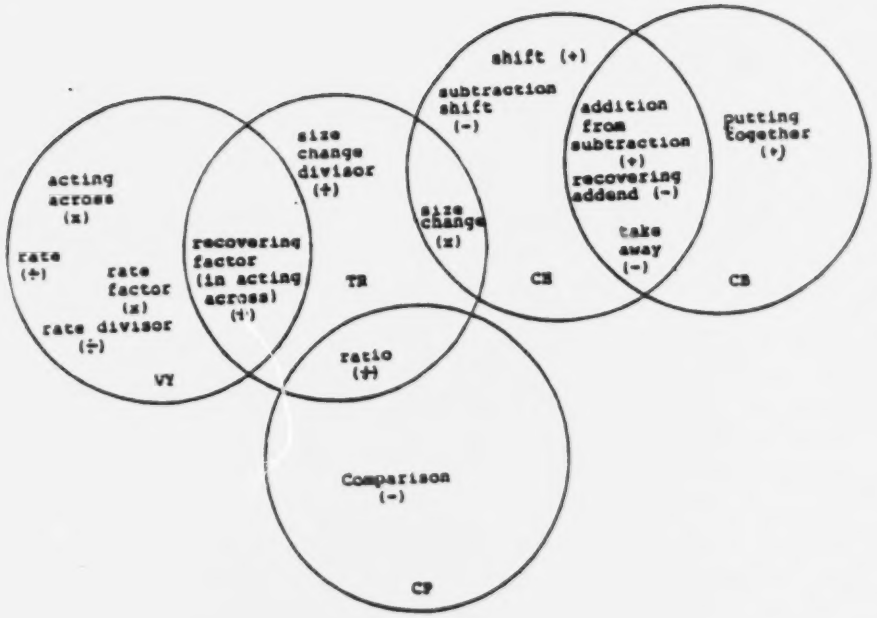


Figure 3: The occurrence of Usiskin and Bell's use classes of arithmetic operations within the five semantic relations. Uses falling in the intersection of two relations can occur in either one. Operations associated with each use are given in parentheses.

- - - - - - - - -

subtraction, or multiplication (as explained previously), and it can demonstrate the addition use of shift or addition from subtraction, the subtraction use of shift or recovering addend, or the multiplication use of size change.

We suspect that Usiskin and Bell's taxonomy will prove to be especially useful in the next phase of our research, the instructional system. In particular, it seems reasonable that their definitions of use will be valuable in explicating the procedural portion of a schema that derives from a particular semantic relation. Thus, we view the schema as broader than either the relation or the use, and it is capable of incorporating both classifications in a meaningful way. The following section describes the nature of a schema and its importance as a general memory structure for semantic relations.

Schema Knowledge: Some Theoretical Considerations

Before describing a theory of how schema knowledge is stored in human memory, we first discuss a general model of memory. Like many other researchers in the field, we posit two types of long-term memory (LTM): declarative and procedural. There seem to be clear distinctions between these two types of memory. Declarative memory contains factual knowledge and knowledge of specific events and experiences. It is usual to assume that this body of knowledge is stored in LTM as one or more semantic networks, linked together with various degrees of association (see for example Anderson, 1983). Through knowledge stored in declarative memory, one can answer questions about who, what, where, and when.

A second type of memory contains skill knowledge rather than factual knowledge. This memory is called procedural, because it consists of sets of general procedures or rules for performing various skills. Unlike declarative memory, procedural memory is highly generalized and not constrained by specific instances or experiences. For example, one uses the same skill of grasping an object with one's fingers in a large number of different situations. The same procedures are utilized, adapting to each situation as necessary. Consequently, one may grasp a pencil, a rock, an apple, or another person's hand without requiring different rules for how to perform each task. A common set of rules applies.

One reason for distinguishing between declarative and procedural knowledge is that the storage mechanisms and retrieval mechanisms seem to be different. Adding knowledge to declarative memory is relatively easy, and there are many "memory tricks" available to help individuals learn declarative facts. For example, individuals can learn a list of unrelated words by encoding them into meaningful sentences (Bower & Clark, 1969). Simple repetition often results in rote learning of declarative information (Rundus, 1971). Encoding such as this, of course, does not insure that the newly acquired knowledge is linked with other, related knowledge.

One may also acquire highly salient declarative knowledge directly without repetition or guided mnemonics. For example, a single experience of an earthquake is often sufficient to establish quite a bit of declarative knowledge about the phenomenon (e.g., noise, shaking).

In contrast, procedural knowledge is difficult to acquire and apparently takes a great deal of practice. As Anderson (1982) points out, many skills take 100 or more hours to acquire. Many motor skills have this

characteristic, for example, learning to type or learning to write. A number of cognitive skills have the same feature, such as learning to multiply, learning to solve physics problems, or learning to program computers.

Given the many distinctions between these two types of LTM knowledge, it becomes important to ask how they are related. Clearly, information of one type calls upon information of the other. By what mechanisms are these two forms of memories united? We suggest that the union occurs to a large extent through the acquisition of a schema.

A schema is a knowledge structure that contains information about how we interact with the environment in a recognizable situation. As such, it contains necessary information about how to recognize the situation and what action(s) we might take under the circumstances. Under this definition, a schema becomes the organizing memory structure that governs our functions in everyday experiences, drawing upon components of both declarative and procedural memories.

An individual is perceived as an active processor of information, using incoming sensory stimuli and previously stored knowledge to make sense of the world. A schema is invoked whenever an individual must formulate a response to his or her environment. The individual's responses in different situations are governed by the schematic knowledge available to the individual. This necessity for a response differentiates a schema from other hypothetical knowledge structures such as plans (Sacerdoti, 1977) or frames (Minsky, 1975) which require no action.

It is reasonably common in artificial intelligence, cognitive science, and cognitive psychological research to define a schema in terms of at least some of the following components: (1) a declarative store of factual knowledge relevant to the schema, (2) a set of conditions that must exist if the schema structure fits the experience, (3) a means of setting goals for satisfying schema constraints, and (4) a set of rules that can be implemented once the schema structure is accepted. We argue here that all four components are necessary.

For any schema, there will be a body of accompanying facts that describe the generic case of the schema. A much-used example is the restaurant schema, for which there are details about definition and structure (e.g., a restaurant is a place where one goes to purchase food, one typically eats the food at the same location, one sits at a table on chairs or benches, and so on). In the specific instance in which the schema is used, more details will be added from the current situation.

A schema will also have a set of conditions that must be met for the schema to apply in any given situation. These are applied to a description of the situation. When the conditions are not met, the schema cannot not be used to explain the situation. These conditions may involve the invocation of other schema structures that are prerequisite to the current one. To continue the example of the restaurant schema, the preconditions are details such as the establishment must serve food, the food must be for sale, there must be chairs, tables, waiters, menus, cooks, etc.

These conditions may be used in either top-down or bottom-up processing. Suppose, for example, that you have entered a building and are standing in a room. You will take in sensory information in order to determine just where you are. If you see several tables at which people are seated on chairs eating a meal, you will likely call upon the restaurant schema to help you interpret all of the details you are processing. If you see many rows of chairs, all facing one direction, you probably search for another schema for clarification, such as a lecture hall or a movie theatre. This is bottom-up processing -- one takes in details and tries to interpret them by means of an existing knowledge structure.

Now consider top-down processing. You desire to go to a restaurant for a meal. You walk into a building. You now look for confirmatory evidence that you have found a restaurant, so you search for details such as tables and chairs, individuals serving food, and so forth. In this case, the knowledge structure directs your attention to specific aspects of the situation. You may not pay attention to the fact that there are paintings on the wall or an orchestra in the corner. These are not central confirmatory conditions for the restaurant schema.

A third feature of schema structure is a mechanism for setting goals in the problem-solving process. In this component reside the rules under which goals are generated, ordered, and satisfied. For example, in the restaurant schema, several associated goals involve deciding what type of restaurant is preferred, how to reach the restaurant, or when to go to the restaurant. The satisfaction of these goals may require additional calls to knowledge found in another schema. For example, if one finds one has no cash, one needs another schema which might be called "how to pay for things without cash". Now information regarding credit cards, checks, or IOU's becomes important as well as the circumstances in which they are reasonably used.

The final aspect of schema structure is a set of rules that governs an individual's response to the situation that

invoked the schema. Thus, once you have recognized that you are in a restaurant (and hence have invoked the restaurant schema), your next action will be directed by conditions that are part of the schema (e.g., you wait to be seated, you order food from the menu, etc.).

Schematic knowledge drives cognitive processing. This function gives the schema a different stature than knowledge previously defined as procedural or declarative. In essence, the schema sits on top of these other LTM structures. In our model, we see a schema as an overlay that encompasses elements of both procedural and declarative knowledge (see Figure 4). As such, it operates as the controlling mechanism in information processing. It determines which procedures and which semantic networks are to be accessed.

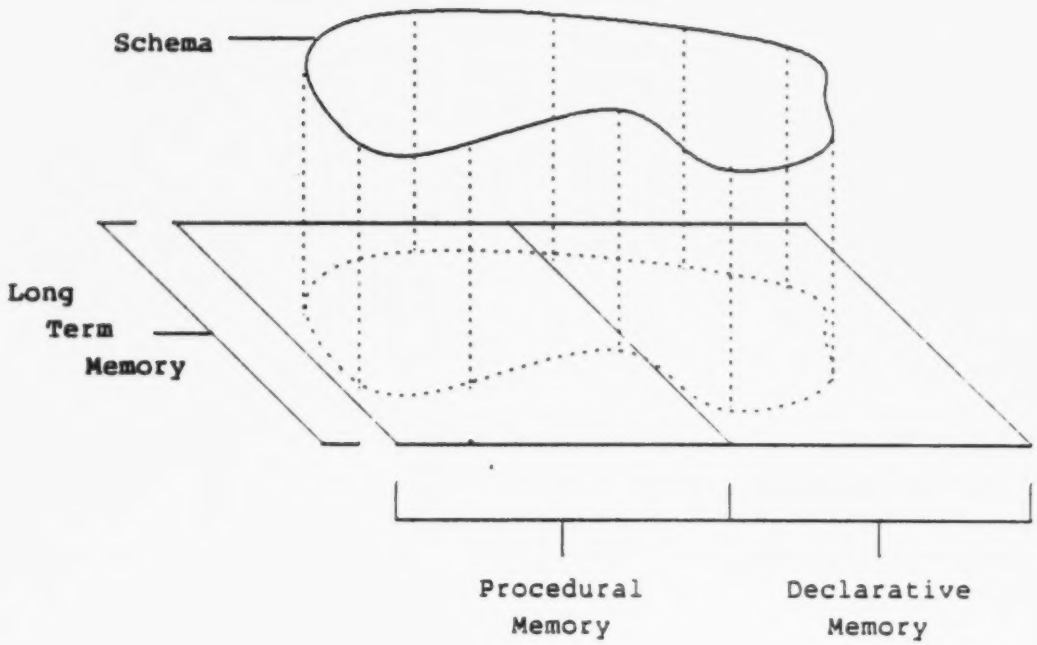
How is a schema accessed and activated? It must direct our recognition processes in a top-down fashion. There are many examples in the research literature of individuals' ability to recognize degraded stimuli, especially when primed. This priming presumably activates a schema skeleton that then directs searching and pattern matching. Degraded stimuli are perceived as being adequate fits only if the schema is successfully activated.

We must also consider bottom-up activation of a schema. If a number of different features of a situation are observed, they may work together to activate the schema skeleton. However, much work in psychology suggests that humans usually attempt to recognize situations and to work in a top-down fashion (cf. Anderson, 1983). This is closely connected to goal-directed behavior. We have expectations about what we expect to experience in our daily lives. Each expectation takes the form of a schema -- so, for example we expect to meet with students in our offices, we expect to open our mailboxes and receive mail. We do not wait until individuals come into a room with us to determine that we are at work and that these are students with questions.

The point about top-down and bottom-up processing is important for understanding learning and instruction. While admitting that we function in primarily a top-down fashion, many psychologists and educators expect learning to be a bottom-up process. That is, a schema is acquired by first solidifying the declarative and procedural components in LTM. Eventually, these elements become interconnected, and a schema skeleton emerges. A major theme of the present research is the challenge of that position: we suggest that schema development may be a top-down process also and that instruction ought to take advantage of this aspect of information processing.

Figure 4

The relationship between schema knowledge, declarative memory, and procedural memory



Related Schema Research

Two alternative views of schema knowledge have their origins in studies of reading and text processing. One has its origins in Bartlett's (1932) study of comprehension and focuses on the nature of stories (Stein, 1982). The second arises from a new theory of cognition called parallel distributive processing (Rumelhart, McClelland and the PDP Research Group, 1986).

Many cognitive psychologists credit Bartlett's (1932) study of text comprehension as the earliest formulation of a schema. Bartlett presented his subjects with a story (the most well-known being "The War of the Ghosts") and asked them to reproduce it. Subjects generally changed and distorted stories according to their own cultural experiences and conventions. Bartlett hypothesized that each subject had an abstract story representation or schema that he or she used to interpret and understand the stories.

Stein (1982) developed a schema-theoretic approach based upon Bartlett's conception of schema. Her research emphasizes the elements of stories and the relationships (such as causal links) that occur. She also examines which elements individuals recall and how the form of a story influences recall. The work of Stein and her colleagues demonstrates the schema nature of stories and the importance of story structure.

Stein's research centers on the structure of stories rather than the organization of memory that individuals must have in order to understand the stories. A primary difference between her work and ours is that her focus is upon the organization of the story and ours is on the organization of the memory processes that are necessary for understanding the story. Consequently, she defines a schema in terms of story features such as setting, different types of episodes, and causal relations. In contrast, we present a general definition of a schema that specifies the type of knowledge contained in the structure and the ways in which that knowledge can be used by information-processing mechanisms. This conception of schema applies to general experiences as well as to stories.

A different schema-theoretic approach has been taken by Rumelhart and his associates (1980, 1986). A central distinction between Rumelhart's view and the one presented in this report is the difference in the conceptualization of long term memory. We adhere to the declarative/procedural model; the PDP group holds a model of connected units.

Rumelhart (1980) developed the following set of schema characteristics: (1) to have variables, (2) to have the capability of being embedded in other schema structures, (3) to represent multiple levels of knowledge from abstract to concrete, (4) to represent knowledge rather than definitions, (5) to be an active process, and (6) to be a recognition device (p. 40-41). This characterization is important in that it specifies what a schema does; it lacks, however, definition of the structure of the schema.

More recently, Rumelhart and his colleagues have developed the notion of a schema within parallel distributive processing theory (Rumelhart et al., 1986; McClelland et al., 1986). Their conception differs greatly from the one presented in this report. The major difference is that a schema is not considered to be a stored memory structure but rather is a collection of activated "units" that become activated simultaneously. As such, a schema need not be the same each time it is required: depending upon each stimulus, some units will be activated and others will not. This approach carries with it major instructional implications. As Rumelhart et al. state,

> There is no point at which it must be decided to to create this or that schema. Learning simply proceeds by connection strength adjustment. . . . (Vol. 2, p. 21)

Our own research suggests that this is not the case: our findings indicate that the teaching of specific schema knowledge leads to efficient learning and problem solving (Marshall, 1987). We provided a group of elementary school children with instruction designed to create a set of schema knowledge structures corresponding to the five semantic relations described previous. The students learned the relations quickly and could differentiate them accurately. We hypothesize that schema knowledge provided the students with a framework for organizing the domain of story problems. In this case, it was necessary to have fixed structures that were learned as such by the students. We will test this hypothesis more thoroughly in the later stages of the present research project.

Schema Knowledge of Semantic Relations

In this and the following section, we define more precisely the form taken by a schema for semantic relations. In this section, the relations are mapped into the four components necessary for a schema. In the next, we discuss a computer simulation of how schema knowledge can be used to solve problems.

The basic schema configurations for the five semantic relations are presented in Figures 5-9 (see pages xx-yy). Each schema is developed with respect to four components: the necessary facts stored in long-term memory, the prerequisites that must exist within the problem for the schema to fit the situation, the goals that may need to be set up, and the rules for using the schema to carry out the needed computations. At this point, we make no suggestion about the order in which this information may be accessed by an individual. The four components of a schema are not necessarily sequential or linear in their acquisition or retrieval. The ordering from one to four is merely for convenience in describing them here.

Declarative Knowledge

The first component contains the declarative knowledge pertinent to the relation. Of primary importance is the representation of a typical problem. We hypothesize the need to store this information in two forms: first, as given in the table, there is a simple story form; second, there is a corresponding graphical structure that represents the same information. The graphs for each schema are given in Figures 6-10. They contain visual information about possible states of a problem and the number of variables that may be required.

The verbal form of the typical problem represents a general template against which the current problem can be examined. It can be used as an analogy: can the current problem be rephrased in such a way that it matches the general case? Similarly, the graphical tree structure represents a second, more specific template. The verbal form allows the individual to approach the problem broadly without paying particular attention to the numbers and names used in the problem. Moving to the tree structure forces the individual to examine the problem in finer detail, mapping the elements of the problem to specific slots of the tree structure. In this way, the individual recognizes which parts of the structure are known, their relationships to each other, and which are yet to be found through arithmetic computation.

For simple problems, an individual may not need to probe declarative knowledge deeply. That is, the structure is readily apparent, and the individual recognizes the components of the problem without going through a formal mapping of problem elements to the graphical template. For more complex problems, the mapping process may elucidate the problem.

Additional details about a relation are also stored as declarative knowledge. These include the number of expected components of the problem, the general characteristics, associated operational uses, and expected operations.

Typically, more declarative knowledge is possessed than can be used in a single situation. When the schema is invoked, some irrelevant features may be activated that do not pertain to the current situation. The activation of the schema may place these elements in working memory. If these do not match elements of the situation, they will be dropped from working memory.

General Prerequisites

A second aspect of schema knowledge is the set of conditions that must be met for the schema to be instantiated. For example, in the Change schema, one condition is that the indicated change be a permanent, physical alteration. In the Combine schema, one must be able to identify the classes that logically comprise a larger group.

The prerequisites serve as a check that the schema is a reasonable one to use under current circumstances. If an individual's construction of the schema does not possess some of them, the schema may be invoked and instantiated inappropriately. If the individual has constructed a schema that entails incorrect prerequisites, he or she may fail to use a schema when it is appropriate.

Goal-Setting Mechanisms

For any problem, the top-level goal is to solve the problem. The solution can be attempted and reached successfully only when all prerequisites have been fulfilled. In some situations, not all prerequisites can be immediately satisfied. This may occur because the problem is ill posed, or it may happen because there are several stages of a problem requiring the solution of subproblems. Whatever the cause, these unsatisfied conditions must be resolved before the schema can be implemented and before actions are carried out.

When prerequisites remain to be met, subgoals are created, and these subgoals must be achieved before the primary goal can be addressed. For example, in a multi-step problem based upon a Change relation, it may be necessary to solve an embedded Vary or Combine problem before reaching the Change solution. Subgoals for solving these embedded relations are created by the goal-setting component of schema knowledge.

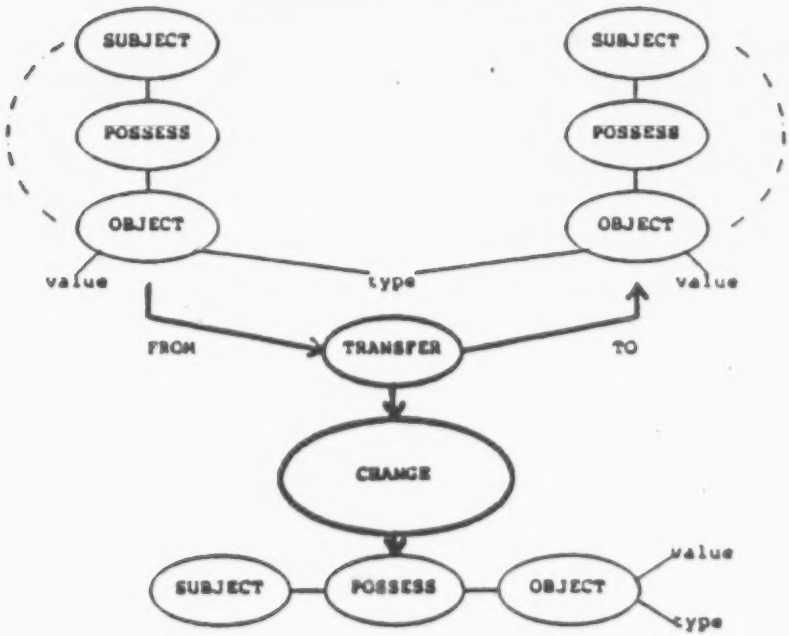
The goal-setting mechanisms are the means by which the subgoals are established and monitored. They recognize which prerequisites are to be satisfied and in which order. They also control acceptance or rejection of goal solutions. Several options are available. On the one hand, it may be possible to satisfy the condition using a variety of default mechanisms. On the other, it may be necessary to invoke another schema or additional aspects of declarative knowledge to gain pertinent information.

Implementation Rules

Once the schema has been successfully invoked and the prerequisites satisfied, it can be used to solve problems. The way in which it is used depends upon knowledge of specific actions that can be taken. These actions are stored as production rules, and they act upon the various components of the situation as defined by declarative knowledge. Depending upon which components of the situation are unknown, specific rules are carried out to determine the value of the component. For most story problems, the actions are applications of arithmetic operations.

Figure 5

The CHANGE Schema



I. DECLARATIVE FACTS

Typical problem: "You have some amount of something, you get more of it (or lose some of it), and now you have a new amount.

Visual tree structure:
- original amount
- amount of change
- result

Characteristics:
- physical alteration
- permanence
- involves time

associated operations:
- addition
- subtraction
- multiplication

associated uses:
- shift
- size change

II. PRECONDITIONS

Can problem be rephrased to match typical problem?

Can tree structure be generated?

Can 2 slots be filled?

If 2 slots are empty, can 1 be constructed from other information?

Is change irreversible?

Are quantities expressed in the same unit of measure?

Can the time periods be identified?

III. GOAL SETTING MECHANISMS

Create goal list

Set top level goal: Solve for a single unknown, which will be 1 of 3 main parts

Determine if problem is multi step

Set subgoals as needed to fill 2 quantity slots

Expect subgoals to point to either VARY or COMBINE: try these first if need to solve subgoals

IV. IMPLEMENTATION RULES

Identify original quantity, transferred quantity, and resulting quantity. (One or more will be unknown.)

Associate quantities with slots

Identify operational use associated with empty slot(s)

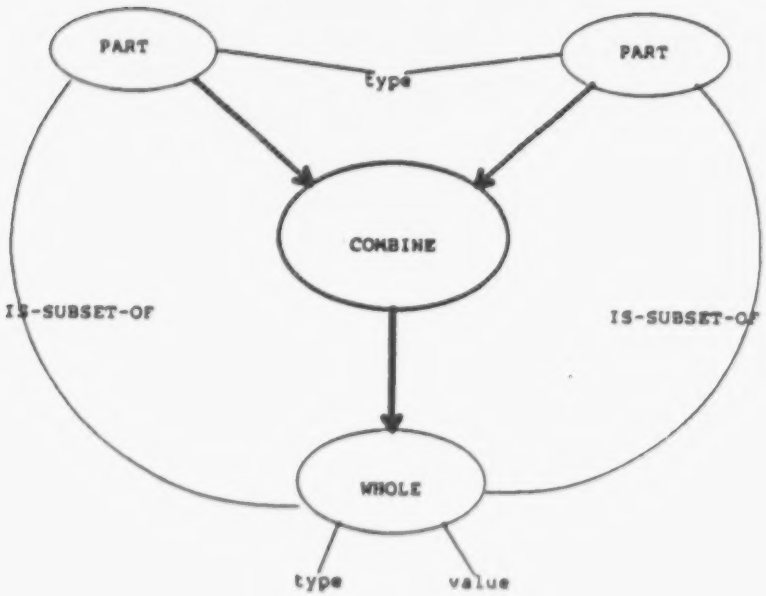
Map from use to particular operation

Carry out operations in order given by goal list

BEST COPY AVAILABLE

Figure 6

The COMBINE Schema



I. DECLARATIVE FACTS

Typical problem: "You have two groups of objects that can be joined into a larger class of objects."

visual tree structure:
- parts
- whole

characteristics:
- hierarchical order
- no action taken
- maintain identities

associated operations:
- addition
- subtraction
- multiplication

associated uses:
- putting together
- taking away

II. PRECONDITIONS

Can problem be rephrased to match typical problem?

Can tree structure be generated?

Can 2 slots be filled?

If 2 slots are empty, can 1 be constructed from other information?

Is part whole relation present?

Do objects maintain identity after the combination?

Can two quantities be expressed as subclasses of the third?

III. GOAL SETTING MECHANISMS

Create goal list

Set top level goal: Solve for a single unknown, which will be 1 of 3 main parts

Determine if problem is multi step

Set subgoals as needed to fill 2 quantity slots

Expect subgoals to point to VARY: try it first if have subgoals

IV. IMPLEMENTATION RULES

Identify hierarchical relationship: Which objects can be combined to form a superordinate category?

Associate quantities with slots

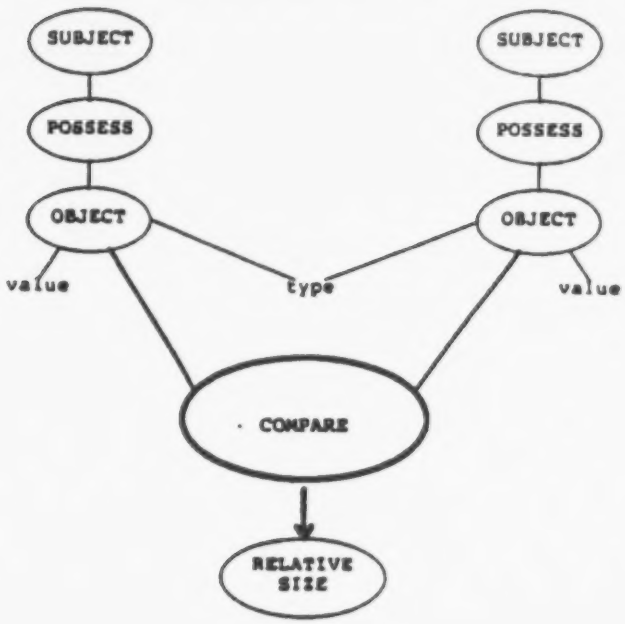
Identify operational use associated with empty slot(s)

Map from use to particular operation

Carry out operations in order given by goal list

Figure 7

The COMPARE Schema



I. DECLARATIVE FACTS

typical problem: "Somebody has more (or less) of something than someone else has. Find out how much more or less."

visual tree structure:
two parts
their difference

characteristics: static
static
no action taken
maintain identities

associated operations:
subtraction
division

associated uses:
comparison
ratio

II. PRECONDITIONS

Can problem be rephrased to match typical problem?

Can tree structure be generated?

Can 2 slots be filled?

If 2 slots are empty, can 1 be constructed from other information?

Are quantities expressed in same unit of measure?

Do objects maintain identity after the comparison?

III. GOAL SETTING MECHANISMS

Create goal list

Set top level goal:
Solve for the unknown which will be the size of the comparison or the identity of the larger or smaller

Determine if problem is multi step

Set subgoals as needed to fill quantity slots

Expect subgoals to point to VARY or COMBINE try these first if have subgoals

IV. IMPLEMENTATION RULES

Identify two quantities that are to be compared

Associate quantities with slots

Identify operational use associated with empty slot(s)

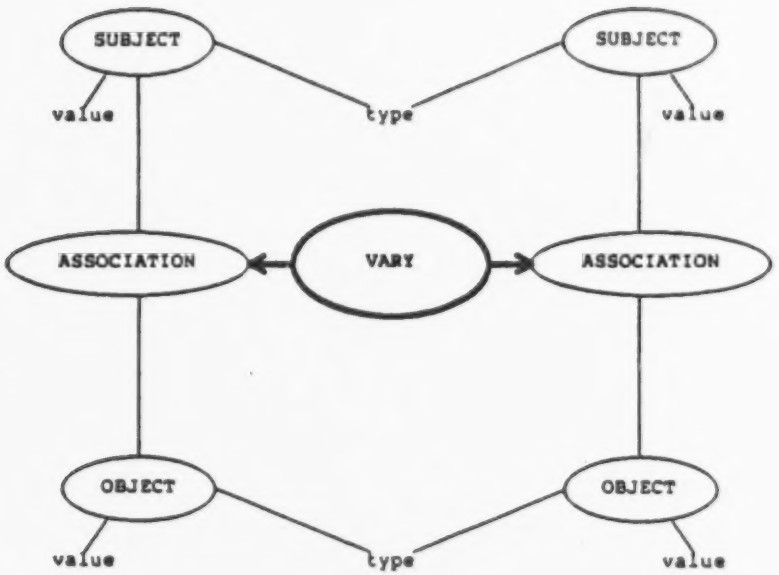
Map from use to particular operation

Carry out operations in order given by goal list

BEST COPY AVAILABLE

Figure 8

The VARY Schema



I. DECLARATIVE FACTS

typical problem: "You have some number of objects. Each has a fixed value. If the number of objects changes, the total value changes also."

visual tree structure:
- number of objects
- fixed value
- new number of objects
- total value

characteristics:
- static
- no action taken
- maintain identities

associated operations:
- multiplication
- division

associated uses:
- acting across
- rate
- factor/divisor
- recovering factor

II. PRECONDITIONS

Can problem be rephrased to match typical problem?

Can tree structure be generated?

Can 3 of 4 slots be filled?

If 2 slots are empty, can 1 be constructed from other information?

Are objects expressed in a common unit of measure?

Are values expressed in a common unit?

Is the function relating a unit and a value the same in all cases?

Is concept of "per unit" expressed? implied?

III. GOAL SETTING MECHANISMS

Create goal list

Set top level goal: Solve for a single unknown, which will be 1 of 4 main parts

Determine if problem is multi step

Set subgoals as needed to fill 3 quantity slots

No expectations about subgoals.

IV. IMPLEMENTATION RULES

Identify objects
Identify per object value
Identify relation between objects and values

Associate quantities with slots

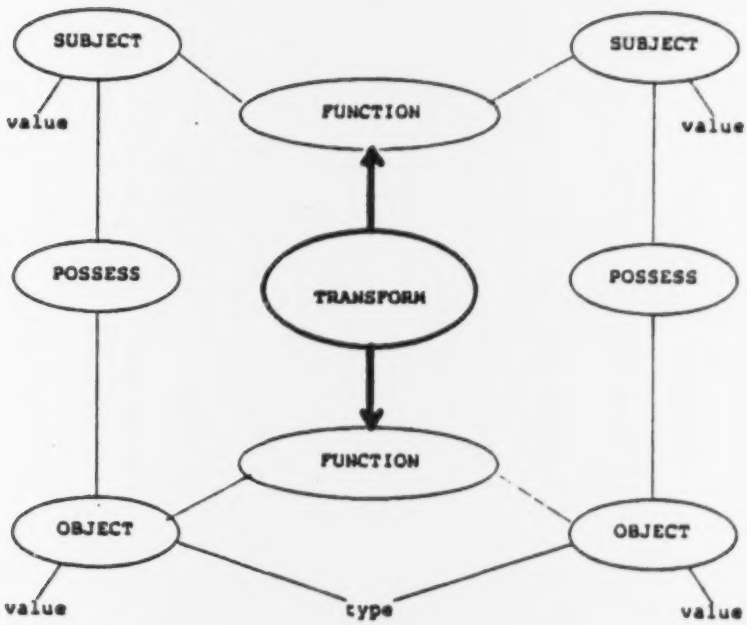
Identify operational use associated with empty slot(s)

Map from use to particular operation

Carry out operations in order given by goal list

Figure 9

The TRANSFORM Schema



I. DECLARATIVE FACTS

typical problem: "Two things are expressed in the same unit of measure. Each of these can be expressed as a function of the other one."

visual tree structure:
- two things
- relational function

characteristics:
- static
- no action taken
- involves relation between 2 objects
- involves part/whole

associated operation:
- addition
- subtraction
- multiplication
- division

associated uses:
- size change
- ratio
- size change
- divisor
- comparison

II. PRECONDITIONS

Can problem be rephrased to match typical problem?

Can tree structure be generated?

Are there 2 distinct things with possible numerical values?

Are the things expressable in the same unit of measure?

Is one of the things stated in terms of the other?

Can one of the things be considered to be a 'whole' so that the other one can be measured against it?

III. GOAL SETTING MECHANISMS

Create goal list

Set top level goal: Solve for a single unknown, which will be 1 of 3 main parts

Determine if problem is multi step

Set subgoals as needed to fill 2 quantity slots

No expectation about secondary steps.

Likely to be the first step in multi step problem

IV. IMPLEMENTATION RULES

Identify the 2 objects to be related.

Identify the function that relates the 2 objects

Identify the object to be expressed in terms of the other

Identify operational use

Map from operational use to operation

Carry out operations in order given by goal list

Computer Modelling of Problem Solving

In this section, we describe computer models of the semantic relations of Change, Combine, Compare, Vary, and Transform. Each is implemented according to the definitions and constraints developed above. The sufficiency of our specifications is evaluated in terms of the computer simulation's success in determining the correct relation for solving a set of simple story problems.

The section has the following outline. First, the form in which story problems are presented to the system is described. Second, the general characteristics of the computer models are given, together with examples of the different semantic relations. Finally, we discuss several issues that remain unresolved.

Propositional Encoding

The creation of a computer system that can parse story problems stated in natural language is beyond the scope of this project. Consequently, we (like others) rely upon propositional encoding of the story problems, and our computer programs operate upon these propositions. The bases for encoding problems into propositions are described here together with examples of several story problems.

Our objective is to represent the semantic relations contained in a story problem and not the individual meanings of words. This is an important point, because it means that the computer model can operate with a reasonable but restricted base of world knowledge. For our purposes, many semantic labels will be indistinguishable. For example, the semantic differences between pieces of furniture such as chair, table, or desk are unimportant. Our system would note only that these are all instances of furniture and are different from each other. The ways in which they are distinct are not (usually) significant factors in solving a story problem. Similarly, the system would know that apples, oranges, lemons, and bananas are all types of fruit. Unless additional characteristics are required by the problem statement (e.g., unless a question such as "how many pieces of yellow fruit are there in the basket"?), they are not represented in the knowledge base.

In like manner, several actions that occur within a story setting have a common meaning. For example, there are many ways to express the notion that an individual possesses something: has, owns, keeps, holds, and grasps are only a few. In most instances, the differences between these terms are not critical to the semantic relation expressed in the problem; they could be interchanged without loss of

understanding. Therefore, it is possible to rely upon a few primitive verbs to express the acts described in story problems. Again, this simplifies the knowledge base used by the computer models.

Each story problem to be solved by the computer simulation is encoded as a set of propositions. These propositions contain all relational and numerical information present in the problem (whether relevant or not). They do not necessarily contain all extraneous details of the situation. For example, consider the problem:

> On her way to work, Mary found $5.00 on the ground. She picked it up and put it in her coat pocket. She already had $16.00. How much money does she have now? (31)

The relevant propositions for this problem need to capture the information that Mary had some money with her and that she acquired some additional money. Several details of the problem are unimportant with respect to the underlying semantic relation. For example, it is not necessary to know that Mary was on her way to work -- her destination has little to do with the structure of the problem. Similarly, it is unimportant in this problem to know in which pocket she put the money.

The General Form of Propositions

Each proposition is an expression composed of five elements:

[subject primitive object direction time]

The subject is the central character or actor of the proposition. It may have a type attribute, a name attribute, and/or an associated numerical value. For example, the subject of a proposition might be "Three boys". In this case, boy is the type, and the numerical value is 3. Name is unassigned because the boys' names are not given. Typically, only the type and/or name is given.

The primitive defines the action of the proposition. As described above, this is generally a class of actions such as possess or transfer.

The object functions in a proposition as a direct object. Like the subject, it may have a type attribute, a name attribute, and/or an associated numerical value. Usually the type attribute and numerical value suffice to describe the object (e.g., 15 cookies, 4 lemons).

The primitive may have directionality. For example, objects may be transferred to or from the subject. The element direction contains this information in two parts. First, the actual direction (to or from) is specified. Second, the recipient of the action and direction is given. The recipient is similar to a indirect object in traditional grammar, but it also allows representation of passive statements. Thus, we can represent "Mary gave John 10 apples" by

[Mary transfer (10 apples) (to: John) ...]

with Mary the subject, transfer the primitive, (10 apples) the object, and (to: John) the direction. In this case, the recipient John is the indirect object of the transfer. Had the problem been stated as "John was given 10 apples by Mary", we have a different proposition:

[John transfer (10 apples) (from: Mary) ...]

in which the recipient contains information about the origin of the transfer.

It is obvious that one could always force passive statements into a propositional form of an active kind (such that the subject always performs the action). Such a policy distorts the structure of the problem. By allowing both passive and active statements through the directionality specification, we preserve as closely as possible the way in which information is presented in a problem.

The fifth element of a proposition is time. This also has two parts, one which denotes in a general way whether the action occurs in the present, past or future, and the second which is a label (such as the day of the week). Verb tenses are encoded in the first part; primitives are always expressed as infinitives.

Types of Propositions

There are three types of propositions: state, event, and query. State propositions reflect a constant state of the world. Two kinds of state propositions may be made. The first describes attributes belonging to the subject. The primitive in this case typically is "possess" and the proposition indicates that a subject has some object. The second kind of state proposition is used to specify group membership and the primitive is "is", such as "George is a boy".

An event proposition denotes a change in possession or some other action in which objects or subjects gain or lose

numerical value. The elements of time and direction are important for event propositions because they define when the event occurred and to whom.

Finally, a query proposition reflects that some element or part of an element is unknown and asks for information about that unknown. In most cases, the unknown is a numerical value associated with an object or subject. Occasionally, the unknown is a particular object or subject (as in the situation where we want only to discover which of two individuals has the most or least of something).

Computer Implementation

The computer models were developed in PRISM, a computer system implemented in InterLispD for use on Xerox D-machines. Created by Pat Langley, PRISM is a system that facilitates construction of production systems within the InterLisp environment (Langley & Neches, 1981; Ohlsson & Langley, 1986). PRISM is especially well-suited for the current project because it distinguishes between working memory and long-term memory and allows generation of alternative architectures for production systems. Thus, it can be altered as necessary for any particular programming task, and we have made modifications to allow representation of the declarative and procedural knowledge used in our simulations.

We define three parts of the system: working memory, procedural memory, and declarative memory. Working memory contains all the information that is active in the system at any given moment. Procedural memory consists of a set of production rules that either identify the underlying relation or take appropriate action once the relation has been specified. Declarative memory is maintained as a semantic network. At this point, the connections or links between elements in the network are primarily those describing inheritance, such as "apple is a fruit" or "boy is a child". We anticipate that other links will be created as we develop a more complete system.

To solve a problem, the system operates only upon working memory (WM), which is initially empty. At various times, it may contain elements from incoming stimuli (e.g., pieces of the problem) or elements from long-term memory that have been activated by production rules.

When a problem is presented, each proposition of the problem enters working memory. As the system encounters the propositions in WM, it checks to see whether certain relationships are present. In order to solve a problem, the system must recognize the semantic relation that underlies

it. Recognition is achieved by the production rules which operate upon the propositions, and it may entail a search in and activation of portions of declarative knowledge. For example, a proposition may contain reference to girls, boys, and children. To determine the relationships that exist among these three categories, the system searches declarative memory and discovers that children is a class of objects that can be decomposed into two subclasses, boys and girls. This information is added to working memory, and it leads to the satisfaction of a primary constraint of the Combine relation (see Table 3).

The recognition set of production rules correspond roughly to the preconditions established for each schematic representation of a semantic relation. These rules are based upon constraint matching rather than upon key words found in the propositions. When the constraints are satisfied, the system "recognizes" the embedded relation in the problem and puts that information in working memory. Thus, in the example above, a statement identifying the Combine relation is added to working memory.

Once a relation has been identified and labeled in working memory, the system activates schema-based rules and attempts to carry out the necessary computations. To do so, it interprets information presented in the propositions with respect to the general framework of the schema that has been invoked. Thus, these rules contain information about the number of quantities that must be already known and how to find values for those that are unknown.

Table 7 contains an example of how the system solves one problem. At the top of the Table, the problem statement is given, followed by a set of propositional encodings. The first three propositions are presented to the system as the original encodings; the last two are created by the system as it solves the problem. The lower portion of the Table illustrates the problem-solving steps that are required for solution by the system. The steps are represented in Table 7 as cycles. Under the current constraints of PRISM, each cycle culminates in the firing of one production rule.

In the first cycle, working memory (WM) contains three propositions: P1, P2, and P3. Several conditions pertain to this situation described by these propositions. First, a transfer of some given amount has taken place and the result of that transfer is unknown. The transfer occurs for a subject (in this case, Sally) and the unknown result also belongs to Sally. The object being transferred is money, expressed in dollars. Finally, no semantic relation has yet been identified, and no schema rules have been called. Only one production rule maps into these conditions. The rule

takes the action of identifying the relation as a CHANGE and places that identification into WM as schema=CH.

In Cycle 2, WM contains the original propositions plus the schema identification. The conditions present in Cycle 1 are also present in Cycle 2, and the constraints to be matched are the same with the exception of the known relation. Again, only one production rule satisfies the constraints. The resulting action is to execute the transfer of P2; that is, a new proposition P4 is created in which the subject now possesses the objects. P4 is added to WM. The original proposition indicating transfer is now obsolete and is deleted from WM.

Following the addition of P4 to WM, the system now recognizes that one subject possesses two known quantities. However, these possessions are recorded at different times. The next cycle checks that one of the possessions occurs earlier than the other and that it can be logically inferred that this possession is unaltered when the second possession takes place. If this is the case (as demonstrated in Cycle 3 of Table 7), the time of the first possession is updated. We represent this updating in proposition P5. The original information is now incorrect and is removed from WM, leaving three propositions: P3, P4, and P5.

In Cycle 4, the system solves for the unknown quantity. The conditions to be met here are that an identified subject possesses two different amounts of an object expressed in a standard unit and that the total of these amounts is unknown. The total is computed and inserted into Proposition 3, replacing the unknown ??? with the computed value. Propositions P4 and P5 are removed immediately from WM and are irretrievable once the aggregation takes place.

Under PRISM architecture, the system continues to run until no production rule can be executed. Thus, the final cycle of any problem-solving endeavor looks like Cycle 5 of Table 7. No productions are acceptable, no constraints are evaluated, and execution terminates.

The existing rules and semantic network are sufficient for solving a set of twenty simple problems we have used in this and in other research about semantic relations (Marshall, 1987). These problems are presented in Table 8. All five semantic relations are present in this set, with four examples of each. The system reaches correct identification of each relation and uses schema knowledge to solve the problems.

Table 7

An Example of How the System Solves a Change Problem

PROBLEM STATEMENT:

Sally had $2.53 in her pocket when she went out for a walk. She found $.35 on the sidewalk and put it in her pocket. How much money did she have at the end of her walk?

PROPOSITIONAL ENCODING:

P1 state proposition:
subject: Sally
primitive: possess
object: money (quantity 2.53) (unit dollar)
direction: nil
time: 0

P2 event proposition:
subject: Sally
primitive: transfer
object: money (quantity .35) (unit dollar)
direction: to Sally
time: 1

P3 state proposition:
subject: Sally
primitive: possess
object: money (quantity ???) (unit dollar)
direction: nil
time: 2

P4 state proposition:
subject: Sally
primitive: possess
object: money (quantity .35) (unit dollar)
direction: nil
time: 1

P5 state proposition:
subject: Sally
primitive: possess
object: money (quantity 2.53) (unit dollar)
direction: nil
time: 1

Table 7 continued

PROBLEM SOLUTION:

CYCLE 1:

Elements of WM:	P1 P2 P3
Constraints satisfied:	transfer occurs (P2 in WM) result of transfer unknown (P3 in WM) subject, objects, and units of objects are constant transfer time earlier than result schema unknown
Action taken:	add schema=CH to WM

Cycle 2:

Elements of WM:	schema=CH P1 P2 P3
Constraints satisfied:	schema=CH transfer occurs (P2 in WM) result of transfer unknown (P3 in WM) transfer time earlier than result
Action taken:	carry out transfer create P4 add P4 to WM delete P2 from WM

CYCLE 3:

Elements of WM:	schema=CH P1 P3 P4
Constraints satisfied:	schema=CH possessions occur at different times
Action taken:	update originial possession time create P5 add P5 to WM delete P1 from WM

Table 7 continued

CYCLE 4:	
Elements of WM:	schema=CH P3 P4 P5
Constraints satisfied:	two possessions occur the times of possession are the same the total possessed is unknown schema=CH
Action taken:	solve for ??? in Proposition (3) modify P3 in WM delete P4 from WM delete P5 from WM
CYCLE 5:	
Elements of WM:	schema=CH P3 [modified]
Constraints satisfied:	none
Action taken:	no production acceptable

Table 8

Examples of Semantic Relations in Arithmetic Story Problems

1. CHANGE RELATION:

For Halloween, Sue put 35 candy bars into a large bowl for children who came "trick-or-treating" at her house. She decided that this might not be enough, so she put another 12 candy bars into the bowl. How many candy bars were in the bowl?

Sally had $2.53 in her pocket when she went out for a walk. She found $.35 on the sidewalk and put it in her pocket. How much money did she have at the end of her walk?

Peter bought 45 cookies for the school party. On the way to school, he got hungry and ate 7 of the cookies. He took the rest of the cookies to school for the party. How many cookies did he contribute to the party?

Before the volleyball game, there were 40 towels for the players to use. After the game, the coach could find only 28 towels. How many towels disappeared during the game?

2. COMBINE RELATION:

At Evans Elementary School, there are 15 members of the boys' basketball team and 18 members of the girls' team. How many students are on basketball teams?

Jodi makes $12.50 a week on her paper route and $3.45 a week for doing chores at home. How much money does Jodi earn each week from these two activities?

At the track meet, there are 83 competitors; 31 of them are boys. How many are girls?

Jerry made fruit salad with apples and bananas. He made 6 1/2 cups of salad. If he put 3 3/4 cups of bananas in the fruit salad, how many cups of apples did he use?

Table 8 continued

3. COMPARE RELATION:

When Chef Jack cooks a roast beef, he bakes it in the oven for 3 hours. When he prepares roast chicken, he only cooks it for 1 1/4 hours. How much longer does a roast beef cook than a roast chicken?

The best gymnast at Central High is Mary. In the last gymnastics meet, she scored 8.1 on the balance beam exercise and 9.4 on the vault. How much better did she do on the vault than on the balance beam?

Mount Ranier, in the state of Washington, is 14,408 feet high. Mount Washington, in New Hampshire, is 6288 feet high. How much higher is Mount Ranier than Mount Washington?

Jeff earns $5.50 per hour at his job, but George only makes $4.25 per hour at his job. How much less per hour does George make?

4. VARY RELATION:

Kevin plays on the school baseball team. Every time any player on his team hits a home run, the coach gives the player 3 baseball cards. Kevin hit 7 home runs this year. How many baseball cards did the coach give Kevin?

One pound of potatoes cost $.35. What would five pounds cost?

Mark's grandfather is 85 years old today. Mark's mother knows that she can't put 85 candles on his birthday cake (because they won't fit). She decides to use 1 candle for every 5 years of Grandfather's age. How many candles should she put on the cake?

Sheila likes to make 3 pitchers of lemonade at once. To do this, she uses 24 cups of water. How much water would she need to make only 1 pitcher of lemonade?

Table 8 continued

5. TRANSFORM RELATION:

In a football-kicking contest, Joe kicked the ball 13 yards farther than Sam kicked it. Joe kicked the ball 30 yards. How far did Sam kick the ball?

Albert spent $3.75 at the school fair. Mike spent 4 times as much as Albert at the fair. How much did Mike spend?

Alice's mother is three times as old as Alice. If her mother is 45 years old, how old is Alice?

Cindy has $4.67. Her friend Bill has $.35 more than Cindy. How much money does Bill have?

Unresolved Issues

We have not yet implemented the goal mechanisms required for full schema representation. The sets of rules described above operate successfully on a small number of multi-step problems we have presented to the system thus far, but they do so by determining which schema can be invoked with current information rather than by identifying the top-level schema of a problem. For example, in a multi-step problem such as the one diagrammed in Figure 8, the current system would not recognize that the problem was essentially a Change relation with other relations embedded in it. Rather, it would first solve a vary problem then look to see if other problems needed to be solved. It would find that it could solve the Combine problem and would do so, still without identifying the need to solve the Change problem. Finally, after solving both the Vary and Combine subproblems, the system would address the Change relation and would carry out the remaining operation and reach a final solution. Thus, the system can solve multi-step problems but it does so without using goals. If information were presented in the problem that could be used to formulate a relation that was actually unnecessary for ultimate solution, the system would be misled and would engage in solving an irrelevant problem.

Most of our attention to date has been on the representations necessary to identify and invoke a single schema. We are now working on the goal-setting mechanisms. We are also engaged in loosening the constraints that identify different relations. This will allow weak identification of a relation and its associated schema and will permit us to begin to model ways in which each schema may be inappropriately instantiated.

Summary

The computer implementation provides support for the schema structures developed here. Using the specifications of the relations and the components of schema knowledge, the computer programs successfully identify and solve a variety of story problems. The next phase of our research will be to develop an instructional environment in which the elements of relational and schematic knowledge can be clearly demonstrated, manipulated, and isolated.

References

Anderson, J. R. (1982). Acquisition of cognitive skill. Psychological Review, 89, 369-406.

Anderson, J. R. (1983). The architecture of cognition. Cambridge, MA: Harvard University Press.

Bartlett, F. C. (1932). Remembering: A study in experimental and social psychology. London: Cambridge University Press.

Bower, G. H. & Clark, M. C. (1969). Narrative stories as mediators for serial learning. Psychonomic Science, 14, 181-182.

Briars, D. J., & Larkin, J. H. (1984). An integrated model of skill in solving elementary word problems. Cognition and Instruction, 1, 245-296.

California Assessment Program (1980). Student Achievement in California Schools, 1979-80 Annual Report, Sacramento: California State Department of Education.

Carpenter, T. P. (1987). Teaching as Problem Solving. Paper presented at the Conference on Teaching and Evaluation of Problem Solving, San Diego, January 1987.

Greeno, J. G., Brown, J. S., Foss, C., Shalin, V., Bee., N. F., Lewis, M. W., and Vitolo, T. M. (1986). Cognitive principles of problem solving and instruction. (Contract No. N-00014-82-K-0613). Washington, DC: Office of Naval Research.

Kintsch, W., & Greeno, J. G. (1985). Understanding and solving word arithmetic problems. Psychological Review,h, 92, 109-129.

Langley, P., & Neches, R. T. (1981). PRISM user's manual. (Technical Report) Computer Science Department, Carnegie-Mellon University, Pittsburgh, PA.

Loftus, E. F. & Suppes, P. (1972). Structural variables that determine problem-solving difficulty in computer-assisted instruction. Journal of Educational Psychology, 63, 531-542.

Marshall, S. P. (1982). Sex differences in solving story problems: A study of strategies and cognitive processes. (Grant No. NIE-G-80-0095). Washington, DC: National Institute of Education.

Marshall, S. P. (1985). An analysis of problem-solving instruction in arithmetic textbooks. Paper presented at the annual meeting of the American Psychological Association, Los Angeles.

Marshall, S. P. (1987). Schema Knowledge for Arithmetic Story Problems: Some issues of instruction, cognition, and application. Unpublished manuscript.

McClelland, J. L., Rumelhart, D. E., & the PDP Research Group (1986). Parallel distributed processing: Explorations in the microstructure of cognition. Volume 2: Psychological and biological models. Cambridge, MA: The MIT Press.

Minsky, M. (1975). A framework for representing knowledge. In P. H. Winston, (Ed.), The psychology of computer vision. New York: McGraw Hill.

Nesher, P., Greeno, J. G., & Riley, M. S. (1982). The development of semantic categories for addition and subtraction. Experimental Studies in Mathematics, 13, 373-394.

Ohlsson, S., & Langley, P. (1986). PRISM tutorial and manual. (Technical Report) Computer Science Department, University of California, Irvine, CA.

Riley, M. S., Greeno, J. G., & Heller, J. I. (1983). Development of children's problem-solving ability in arithmetic. In H. P. Ginsburg (Ed.), The development of mathematical thinking. New York: Academic Press.

Rumelhart, D. E. (1980). Schemata: The building blocks of cognition. In R. Spiro, D. Bruce, & W. Brewer (Eds.), Theoretical issues in reading comprehension. Hillsdale, NJ: Erlbaum.

Rumelhart, D. E., McClelland, J. L., & the PDP Research Group (1986). Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations. Cambridge, MA: The MIT Press.

Rundus, D. (1971). Analysis of rehearsal processes in free recall. Journal of Experimental Psychology, 89, 63-77.

Sacerdoti, E. D. (1977). A structure for plans and behavior. New York: Elsevier North-Holland.

Stein, N., & Trabasso, T. (1982). What's in a story? An approach to to comprehension and instruction. In R. Glaser, (Ed.), Advances in instructional psychology. Volume 2. Hillsdale, NJ: Erlbaum.

Usiskin, Z., & Bell, M. (1983). Applying arithmetic: A handbook of applications of arithmetic. Part II: Operations. Arithmetics and Its Application Project, Department of Education, The University of Chicago.

DISTRIBUTION LIST

Dr. Beth Adelson
Department of Computer Science
Tufts University
Medford, MA 02155

Air Force Human Resources Lab
AFHRL/MPD
Brooks AFB, TX 78235

Dr. Robert Ahlers
Code N711
Human Factors Laboratory
Naval Training Systems Center
Orlando, FL 32813

Dr. Ed Aiken
Navy Personnel R&D Center
San Diego, CA 92152-6800

Dr. John R. Anderson
Department of Psychology
Carnegie-Mellon University
Pittsburgh, PA 15213

Dr. Nancy S. Anderson
Department of Psychology
University of Maryland
College Park, MD 20742

Technical Director, ARI
5001 Eisenhower Avenue
Alexandria, VA 22333

Dr. Patricia Baggett
University of Colorado
Department of Psychology
Box 345
Boulder, CO 80309

Dr. Eva L. Baker
UCLA Center for the Study
of Evaluation
145 Moore Hall
University of California
Los Angeles, CA 90024

Dr. William M. Bart
University of Minnesota
Dept. of Educ. Psychology
330 Burton Hall
178 Pillsbury Dr., S.E.
Minneapolis, MN 55455

Dr. Lee Roy Beach
Dept. of Psychology (NI-25)
University of Washington
Seattle, WA 98195

Dr. Isaac Bejar
Educational Testing Service
Princeton, NJ 08450

Dr. Mark H. Bickhard
University of Texas
EDB 504 ED Psych
Austin, TX 78712

Dr. John Black
Teachers College
Columbia University
525 West 121st Street
New York, NY 10027

Dr. R. Darrell Bock
University of Chicago
NORC
6030 South Ellis
Chicago, IL 60637

Dr. J. C. Boudreaux
Center for Manufacturing
Engineering
National Bureau of Standards
Gaithersburg, MD 20899

Dr. Jomills H. Braddock II
Center for the Social
Organization of Schools
The Johns Hopkins University
3505 North Charles Street
Baltimore, MD 21218

Dr. Robert Breaux
Code N-095R
Naval Training Systems Center
Orlando, FL 32813

Commanding Officer
CAPT Lorin W. Brown
NROTC Unit
Illinois Institute of Tech.
3300 S. Federal Street
Chicago, IL 60616-3793

Dr. John S. Brown
XEROX Palo Alto Research
Center
3333 Coyote Road
Palo Alto, CA 94304

Dr. John Bruer
The James S. McDonnell
Foundation
University Club Tower, Suite 1610
1034 South Brentwood Blvd.
St. Louis, MO 63117

Dr. Patricia A. Butler
OERI
555 New Jersey Ave., NW
Washington, DC 20208

Joanne Capper
Center for Research into Practice
1718 Connecticut Ave., N.W.
Washington, DC 20009

Dr. Susan Carey
Harvard Graduate School of
Education
337 Gutman Library
Appian Way
Cambridge, MA 02138

Dr. Pat Carpenter
Carnegie-Mellon University
Department of Psychology
Pittsburgh, PA 15213

Dr. John B. Carroll
409 Elliott Rd.
Chapel Hill, NC 27514

LCDR Robert Carter
Office of the Chief
of Naval Operations
OP-01B
Pentagon
Washington, DC 20350-2000

Dr. Davida Charney
English Department
Penn State University
University Park, PA 16802

Dr. Paul R. Chatelier
OUSDRE
Pentagon
Washington, DC 20350-2000

Dr. Michelene Chi
Learning R & D Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15213

Dr. William Clancey
Stanford University
Knowledge Systems Laboratory
701 Welch Road, Bldg. C
Palo Alto, CA 94304

Dr. Paul Cobb
Purdue University
Educational Building
W. Lafayette, IN 47907

Dr. Allan M. Collins
Bolt Beranek & Newman, Inc.
50 Moulton Street
Cambridge, MA 02138

Dr. Stanley Collyer
Office of Naval Technology
Code 222
800 N. Quincy Street
Arlington, VA 22217-5000

Dr. Jere Confrey
Cornell University
Dept. of Education
Room 490 Roberts
Ithaca, NY 14853

Dr. William Crano
Department of Psychology
Texas A&M University
College Station, TX 77843
USA

Dr. Mary Cross
Department of Education
Adult Literacy Initiative
Room 4145
400 Maryland Avenue, SW
Washington, DC 20202

Dr. Cary Czichon
Intelligent Instructional Systems
Texas Instruments AI Lab
P.O. Box 660245
Dallas, TX 75266

Brian Dallman
3400 TTW/TTGXS
Lowry AFB, CO 80230-5000

Dr. Robert B. Davis
Curriculum Laboratory
(Education)
University of Illinois
Urbana, IL 61801

LT John Deaton
ONR Code 125
800 N. Quincy Street
Arlington, VA 22217-5000

Dr. Natalie Dehn
Department of Computer and
Information Science
University of Oregon
Eugene, OR 97403

Dr. Sharon Derry
Florida State University
Department of Psychology
Tallahassee, FL 32306

Dr. Stephanie Doan
Code 6021
Naval Air Development Center
Warminster, PA 18974-5000

Dr. J. S. Donn
Cornell University
Dept. of Education
Roberts Hall
Ithaca, NY 14853

Defense Technical
Information Center
Cameron Station, Bldg 5
Alexandria, VA 22314
Attn: TC
(12 Copies)

Dr. Richard Duran
University of California
Santa Barbara, CA 93106

Dr. Susan Embretson
University of Kansas
Psychology Department
426 Fraser
Lawrence, KS 66045

Dr. Randy Engle
Department of Psychology
University of South Carolina
Columbia, SC 29208

ERIC Facility-Acquisitions
4833 Rugby Avenue
Bethesda, MD 20014

Dr. Beatrice J. Farr
Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

Dr. Marshall J. Farr
Farr-Sight Co.
2520 North Vernon Street
Arlington, VA 22207

Dr. Pat Federico
Code 511
NPRDC
San Diego, CA 92152-6800

Dr. Paul Feltovich
Southern Illinois University
School of Medicine
Medical Education Department
P.O. Box 3926
Springfield, IL 62708

Mr. Wallace Feurzeig
Educational Technology
Bolt Beranek & Newman
10 Moulton St.
Cambridge, MA 02238

Dr. Gerhard Fischer
Liebiggasse 5/3
A 1010 Vienna
AUSTRIA

Dr. Gerhard Fischer
University of Colorado
Department of Computer Science
Boulder, CO 80309

Dr. Linda Flower
Carnegie-Mellon University
Department of English
Pittsburgh, PA 15213

Dr. Barbara A. Fox
University of Colorado
Department of Linguistics
Boulder, CO 80309

Dr. Carl H. Frederiksen
McGill University
3700 McTavish Street
Montreal, Quebec H3A 1Y2
CANADA

Dr. John R. Frederiksen
Bolt Beranek & Newman
50 Moulton Street
Cambridge, MA 02138

Dr. Norman Frederiksen
Educational Testing Service
Princeton, NJ 08541

Dr. Michael Friendly
Psychology Department
York University
Toronto ONT
CANADA M3J 1P3

Dr. Robert M. Gagne
1456 Mitchell Avenue
Tallahassee, FL 32303

Dr. Dedre Gentner
University of Illinois
Department of Psychology
603 E. Daniel St.
Champaign, IL 61820

Chair, Department of
Psychology
George Mason University
Fairfax, VA 22030

Dr. Robert D. Gibbons
University of Illinois-Chicago
P.O. Box 6998
Chicago, IL 69680

Dr. Lee Giles
AFOSR
Bolling AFB
Washington, DC 20332

Dr. Herbert Ginsburg
Teachers College
Columbia University
525 West 121st Street
New York, NY 10027

Lee Gladwin
Route 3 -- Box 225
Winchester, VA 22601

Dr. Robert Glaser
Learning Research
& Development Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15260

Dr. Arthur M. Glenberg
University of Wisconsin
W. J. Brogden Psychology Bldg.
1202 W. Johnson Street
Madison, WI 53706

Dr. Marvin D. Glock
13 Stone Hall
Cornell University
Ithaca, NY 14853

Dr. Sam Glucksberg
Department of Psychology
Princeton University
Princeton, NJ 08540

Dr. Susan Goldman
University of California
Santa Barbara, CA 93106

Dr. Sherrie Gott
AFHRL/MODJ
Brooks AFB, TX 78235

Dr. Wayne Gray
Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

Dr. Bert Green
Johns Hopkins University
Department of Psychology
Charles & 34th Street
Baltimore, MD 21218

Dr. James G. Greeno
University of California
Berkeley, CA 94720

Prof. Edward Haertel
School of Education
Stanford University
Stanford, CA 94305

Dr. Henry M. Halff
Halff Resources, Inc.
4918 33rd Road, North
Arlington, VA 22207

Dr. Ronald K. Hambleton
Prof. of Education & Psychology
University of Massachusetts
at Amherst
Hills House
Amherst, MA 01003

Dr. Ray Hannapel
Scientific and Engineering
Personnel and Education
National Science Foundation
Washington, DC 20550

Prof. John R. Hayes
Carnegie-Mellon University
Department of Psychology
Schenley Park
Pittsburgh, PA 15213

Dr. Barbara Hayes-Roth
Department of Computer Science
Stanford University
Stanford, CA 95305

Dr. Shirley Brice Heath
School of Education
Stanford University
Stanford, CA 94305

Dr. Joan I. Heller
505 Haddon Road
Oakland, CA 94606

Dr. Shelly Heller
Department of Electrical Engineering & Computer Science
George Washington University
Washington, DC 20052

Dr. Jim Hollan
Intelligent Systems Group
Institute for
Cognitive Science (C-015)
UCSD
La Jolla, CA 92093

Dr. Melissa Holland
Army Research Institute for
Behavioral and Social Sciences
5001 Eisenhower Avenue
Alexandria, VA 22333

Dr. Robert W. Holt
Department of Psychology
George Mason University
4400 University Drive
Fairfax, VA 22030

Dr. Thomas Holzman
Lockheed Georgia
Dept. 64-31
Zone 278
Marietta, GA 30063

Ms. Julia S. Hough
Lawrence Erlbaum Associates
6012 Greene Street
Philadelphia, PA 19144

Dr. James Howard
Dept. of Psychology
Human Performance Laboratory
Catholic University of
America
Washington, DC 20064

Dr. Lloyd Humphreys
University of Illinois
Department of Psychology
603 East Daniel Street
Champaign, IL 61820

Dr. Ed Hutchins
Intelligent Systems Group
Institute for
Cognitive Science (C-015)
UCSD
La Jolla, CA 92093

Dr. Huynh Huynh
College of Education
Univ. of South Carolina
Columbia, SC 29208

Dr. Dillon Inouye
WICAT Education Institute
Provo, UT 84057

Dr. Alice Isen
Department of Psychology
University of Maryland
Catonsville, MD 21228

Dr. Robert Jannarone
Department of Psychology
University of South Carolina
Columbia, SC 29208

Dr. Claude Janvier
Directeur, CIRADE
Universite' du Quebec a Montreal
P.O. Box 8888, St. "A"
Montreal, Quebec H3C 3P8
CANADA

Dr. Robin Jeffries
Hewlett-Packard Laboratories
P.O. Box 10490
Palo Alto, CA 94303-0971

CDR Tom Jones
ONR Code 125
800 N. Quincy Street
Arlington, VA 22217-5000

Dr. Douglas H. Jones
Thatcher Jones Associates
P.O. Box 6640
10 Trafalgar Court
Lawrenceville, NJ 08648

Dr. Marcel Just
Carnegie-Mellon University
Department of Psychology
Schenley Park
Pittsburgh, PA 15213

Dr. Ruth Kanfer
University of Minnesota
Department of Psychology
Elliott Hall
75 E. River Road
Minneapolis, MN 55455

Dr. Milton S. Katz
Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

Dr. Frank Keil
Department of Psychology
Cornell University
Ithaca, NY 14850

Dr. Wendy Kellogg
IBM T. J. Watson Research Ctr.
P.O. Box 218
Yorktown Heights, NY 10598

Dr. Dennis Kibler
University of California
Department of Information
and Computer Science
Irvine, CA 92717

Dr. David Kieras
University of Michigan
Technical Communication
College of Engineering
1223 E. Engineering Building
Ann Arbor, MI 48109

Dr. Paula Kirk
Oakridge Associated Univ.
University Programs Division
P.O. Box 117
Oakridge, TN 37831-0117

Dr. Kenneth Kotovsky
Department of Psychology
Community College of
Allegheny County
800 Allegheny Avenue
Pittsburgh, PA 15233

Dr. Pat Langley
University of California
Department of Information
and Computer Science
Irvine, CA 92717

Dr. Jill Larkin
Carnegie-Mellon University
Department of Psychology
Pittsburgh, PA 15213

Dr. Jean Lave
School of Social Sciences
University of California
Irvine, CA 92717

Dr. R. W. Lawler
ARI 6 S 10
5001 Eisenhower Avenue
Alexandria, VA 22333-5600

Dr. Alan M. Lesgold
Learning Research and
Development Center
University of Pittsburgh
Pittsburgh, PA 15260

Dr. Alan Leshner
Deputy Division Director
Behavioral and Neural Sciences
National Science Foundation
1800 G Street
Washington, DC 20550

Dr. John Levine
Learning R&D Center
University of Pittsburgh
Pittsburgh, PA 15260

Dr. Michael Levine
Educational Psychology
210 Education Bldg.
University of Illinois
Champaign, IL 61801

Dr. Clayton Lewis
University of Colorado
Department of Computer Science
Campus Box 430
Boulder, CO 80309

Matt Lewis
Department of Psychology
Carnegie-Mellon University
Pittsburgh, PA 15213

Library
Naval War College
Newport, RI 02940

Library
Naval Training Systems Center
Orlando, FL 32813

Dr. Marcia C. Linn
Lawrence Hall of Science
University of California
Berkeley, CA 94720

Dr. Robert Linn
College of Education
University of Illinois
Urbana, IL 61801

Dr. Sandra P. Marshall
Dept. of Psychology
San Diego State University
San Diego, CA 92182

Dr. Richard E. Mayer
Department of Psychology
University of California
Santa Barbara, CA 93106

Dr. James McBride
Psychological Corporation
c/o Harcourt, Brace,
Javanovich Inc.
1250 West 6th Street
San Diego, CA 92101

Dr. Joe McLachlan
Navy Personnel R&D Center
San Diego, CA 92152-6800

Dr. James S. McMichael
Navy Personnel Research
and Development Center
Code 05
San Diego, CA 92152

Dr. Barbara Means
Human Resources
Research Organization
1100 South Washington
Alexandria, VA 22314

Dr. Douglas L. Medin
Department of Psychology
University of Illinois
603 E. Daniel Street
Champaign, IL 61820

Dr. Arthur Melmed
U. S. Department of Education
724 Brown
Washington, DC 20208

Dr. George A. Miller
Department of Psychology
Green Hall
Princeton University
Princeton, NJ 08540

Dr. Lance Miller
IBM-FSD Headquarters
6600 Rockledge Drive
Bethesda, MD 20817

Dr. Andrew R. Molnar
Scientific and Engineering
Personnel and Education
National Science Foundation
Washington, DC 20550

Dr. William Montague
NPRDC Code 13
San Diego, CA 92152-6800

Dr. Nancy Morris
Search Technology, Inc.
5550-A Peachtree Parkway
Technology Park/Summit
Norcross, GA 30092

Dr. Randy Mumaw
Program Manager
Training Research Division
HumRRO
1100 S. Washington
Alexandria, VA 22314

Dr. Richard E. Nisbett
University of Michigan
Institute for Social Research
Room 5261
Ann Arbor, MI 48109

Dr. Donald A. Norman
Institute for Cognitive
Science C-015
U. C. San Diego
La Jolla, California 92093

Director, Training Laboratory,
NPRDC (Code 05)
San Diego, CA 92152-6800

Director, Manpower and
Personnel Laboratory,
NPRDC (Code 06)
San Diego, CA 92152-6800

Director, Human Factors
& Organizational Systems Lab,
NPRDC (Code 07)
San Diego, CA 92152-6800

Dr. Harold F. O'Neil, Jr.
School of Education - WPH 801
Department of Educational
Psychology & Technology
Univ. of Southern California
Los Angeles, CA 90089-0031

Dr. Michael Oberlin
Naval Training Systems Center
Code 711
Orlando, FL 32813-7100

Dr. Stellan Ohlsson
Learning R & D Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15213

Office of Naval Research,
Code 1142CS
800 N. Quincy Street
Arlington, VA 22217-5000
(6 Copies)

Office of Naval Research,
Code 125
800 N. Quincy Street
Arlington, VA 22217-5000

Psychologist
Office of Naval Research
Branch Office, London
Box 39
FPO New York, NY 09510

Special Assistant for Marine
Corps Matters,
ONR Code 00MC
800 N. Quincy St.
Arlington, VA 22217-5000

Psychologist
Office of Naval Research
Liaison Office, Far East
APO San Francisco, CA 96503

Office of Naval Research,
Resident Representative,
UCSD
University of California,
San Diego
La Jolla, CA 92093-0001

Dr. Judith Orasanu
Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

Dr. Roy Pea
Bank Street College of
Education
610 W. 112th Street
New York, NY 10025

Dr. Virginia E. Pendergrass
Code 711
Naval Training Systems Center
Orlando, FL 32813-7100

Dr. Nancy Pennington
University of Chicago
Graduate School of Business
1101 E. 58th St.
Chicago, IL 60637

Dr. Ray Perez
ARI (PERI-II)
5001 Eisenhower Avenue
Alexandria, VA 2233

Dr. David N. Perkins
Educational Technology Center
337 Gutman Library
Appian Way
Cambridge, MA 02138

Dr. Tjeerd Plomp
Twente University of Tech.
Department of Education
P.O. Box 217
7500 AE ENSCHEDE
THE NETHERLANDS

Dr. Martha Polson
Department of Psychology
Campus Box 346
University of Colorado
Boulder, CO 80309

Dr. Harry E. Pople
University of Pittsburgh
Decision Systems Laboratory
1360 Scaife Hall
Pittsburgh, PA 15261

Dr. Joseph Psotka
ATTN: PERI-1C
Army Research Institute
5001 Eisenhower Ave.
Alexandria, VA 22333

Dr. Lynne Reder
Department of Psychology
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

Dr. James A. Reggia
University of Maryland
School of Medicine
Department of Neurology
22 South Greene Street
Baltimore, MD 21201

Dr. Wesley Regian
AFHRL/MOD
Brooks AFB, TX 78235

Dr. Fred Reif
Physics Department
University of California
Berkeley, CA 94720

Dr. Lauren Resnick
Learning R & D Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15213

Dr. Gil Ricard
Mail Stop C04-14
Grumman Aerospace Corp.
Bethpage, NY 11714

Mark Richer
1041 Lake Street
San Francisco, CA 94118

Dr. Mary S. Riley
Program in Cognitive Science
Center for Human Information Processing
University of California
La Jolla, CA 92093

Dr. Linda G. Roberts
Science, Education, and Transportation Program
Office of Technology Assessment
Congress of the United States
Washington, DC 20510

Dr. William B. Rouse
Search Technology, Inc.
5550-A Peachtree Parkway
Technology Park/Summit
Norcross, GA 30092

Dr. Fumiko Samejima
Department of Psychology
University of Tennessee
Knoxville, TN 37916

Dr. Robert Sasmor
HQDA DAMA-ARL
Pentagon, Room 3E516
Washington, DC 20310-0631
USA

Dr. Alan H. Schoenfeld
University of California
Department of Education
Berkeley, CA 94720

Dr. Janet Schofield
Learning R&D Center
University of Pittsburgh
Pittsburgh, PA 15260

Dr. Judith Segal
OERI
555 New Jersey Ave., NW
Washington, DC 20208

Dr. Colleen M. Seifert
Intelligent Systems Group
Institute for Cognitive Science (C-015)
UCSD
La Jolla, CA 92093

Dr. Daniel Sewell
Search Technology, Inc.
5550-A Peachtree Parkway
Technology Park/Summit
Norcross, GA 30092

Dr. Sylvia A. S. Shafto
Department of Computer Science
Towson State University
Towson, MD 21204

Dr. Lee Shulman
Stanford University
1040 Cathcart Way
Stanford, CA 94305

Dr. Valerie Shute
AFHRL/MOE
Brooks AFB, TX 78235

Dr. Robert S. Siegler
Carnegie-Mellon University
Department of Psychology
Schenley Park
Pittsburgh, PA 15213

Dr. Edward Silver
Dept. of Mathematics
San Diego State University
San Diego, CA 92182

LTCOL Robert Simpson
Defense Advanced Research
Projects Administration
1400 Wilson Blvd.
Arlington, VA 22209

Dr. Zita M Simutis
Instructional Technology
Systems Area
ARI
5001 Eisenhower Avenue
Alexandria, VA 22333

Dr. Derek Sleeman
Dept. of Computing Science
King's College
Old Aberdeen
AB9 2UB
UNITED KINGDOM

Dr. Richard E. Snow
Department of Psychology
Stanford University
Stanford, CA 94306

Dr. Elliot Soloway
Yale University
Computer Science Department
P.O. Box 2158
New Haven, CT 06520

Dr. Richard Sorensen
Navy Personnel R&D Center
San Diego, CA 92152-6800

Dr. Paul Speckman
University of Missouri
Department of Statistics
Columbia, MO 65201

Dr. Robert Sternberg
Department of Psychology
Yale University
Box 11A, Yale Station
New Haven, CT 06520

Dr. Thomas Sticht
Navy Personnel R&D Center
San Diego, CA 92152-6800

Mr. Brad Sympson
Navy Personnel R&D Center
San Diego, CA 92152-6800

Dr. John Tangney
AFOSR/NL
Bolling AFB, DC 20332

Dr. Kikumi Tatsuoka
CERL
252 Engineering Research
Laboratory
Urbana, IL 61801

Dr. Martin M. Taylor
DCIEM
Box 2000
Downsview, Ontario
CANADA

Dr. Robert P. Taylor
Teachers College
Columbia University
New York, NY 10027

Dr. Robert Tsutakawa
University of Missouri
Department of Statistics
222 Math. Sciences Bldg.
Columbia, MO 65211

Dr. Zita E. Tyer
Department of Psychology
George Mason University
4400 University Drive
Fairfax, VA 22030

Headquarters, U. S. M. C.
Code MPI-20
Washington, DC 20380

Dr. Kurt Van Lehn
Department of Psychology
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

Dr. Jerry Vogt
Navy Personnel R&D Center
Code 51
San Diego, CA 92152-6800

Dr. Ming-Mei Wang
Lindquist Center
for Measurement
University of Iowa
Iowa City, IA 52242

Dr. Beth Warren
Bolt Beranek & Newman, Inc.
50 Moulton Street
Cambridge, MA 02138

Dr. Heather Wild
Naval Air Development
Center
Code 6021
Warminster, PA 18974-5000

Dr. William Clancey
Stanford University
Knowledge Systems Laboratory
701 Welch Road, Bldg. C
Palo Alto, CA 94304

Dr. Hilda Wing
Army Research Institute
5001 Eisenhower Ave.
Alexandria, VA 22333

Ms. Marilyn Wingersky
Educational Testing Service
Princeton, NJ 08541

Dr. Robert A. Wisher
U.S. Army Institute for the
Behavioral and Social Sciences
5001 Eisenhower Avenue
Alexandria, VA 22333

Dr. Martin F. Wiskoff
Navy Personnel R & D Center
San Diego, CA 92152-6800

Mr. John H. Wolfe
Navy Personnel R&D Center
San Diego, CA 92152-6800

Dr. Dan Wolz
AFHRL/MOE
Brooks AFB, TX 78235

Dr. George Wong
Biostatistics Laboratory
Memorial Sloan-Kettering
Cancer Center
1275 York Avenue
New York, NY 10021

Dr. Wallace Wulfeck, III
Navy Personnel R&D Center
San Diego, CA 92152-6800

Dr. Joe Yasatuke
AFHRL/LRT
Lowry AFB, CO 80230

Dr. Masoud Yazdani
Dept. of Computer Science
University of Exeter
Exeter EX4 4QL
Devon, ENGLAND

Dr. Joseph L. Young
Memory & Cognitive
Processes
National Science Foundation
Washington, DC 20550